ตะลุยโจทย์ PYTHON 500 ข้อ

พร้อมเฉลยอย่างละเอียด

ผู้เขียน

ฆฤณ ชินประสาทศักดิ์

นันทนัช ฟูสามป๊อก

พนมวัฒน์ วรัทเศรษฐ์

พิเศษ!
VDO สอน PYTHON
30
ชั่วโมง

TAUTOLOGY

ตะลุยโจทย์ Python 500 ข้อ
พร้อมเฉลยอย่างละเอียด

ผู้เขียน ฆฤณ ชินประสาทศักดิ์
นันทนัช ฟูสามปือก
พนมวัฒน์ วรัทเศรษฐ์

ออกแบบปกและรูปเล่ม ณิชชา คงสีดี

พิมพ์ครั้งที่ 1 มีนาคม 2564

ข้อมูลทางบรรณานุกรมของหอสมุดแห่งชาติ

ฆฤณ ชินประสาทศักดิ์.
ตะลุยโจทย์ Python 500 ข้อ พร้อมเฉลยอย่างละเอียด.– นนทบุรี : เมดบายเอไอ, 2564. 420 หน้า

1. ไพธอน(ภาษาคอมพิวเตอร์) – ข้อสอบและเฉลย. I. นันทนัช ฟูสามปือก, ผู้แต่งร่วม. II. พนมวัฒน์ วรัทเศรษฐ์, ผู้แต่งร่วม. III. ชื่อเรื่อง

005.133
ISBN 978-616-93753-0-2
ราคา 495 บาท

จัดทำโดย บริษัท เมดบายเอไอ จำกัด
49/123 หมู่ที่ 9 ตำบลบางตลาด
อำเภอปากเกร็ด จ.นนทบุรี
โทรศัพท์ 086-524-4463

จัดจำหน่ายโดย บริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน)
1858/87-90 ถ.เทพรัตน
แขวงบางนาใต้ เขตบางนา กรุงเทพฯ 10260
โทรศัพท์ 0-2826-8000, 0-2826-8000
โทรสาร 0-2836-8356-9
http://www.se-ed.com

ติดต่อสอบถามเพิ่มเติม https://madebyai.io
https://www.facebook.com/tautologyai

พิมพ์ที่ บริษัท วัน โอ ไฟว์ ดิจิตอลพริ้นติ้ง จำกัด เลขที่ 89/9 หมู่ 9 ต.บางแก้ว อ.บางพลี จ.สมุทรปราการ 10540

Sometimes it's the people no one imagines anything of who do the things that no one can imagine.

...บางครั้ง คนที่ไม่มีใครคาดคิด
คือคนที่ทำสิ่งที่ไม่มีใครคาดฝันได้

——— ALAN TURING ———

คำนำ

Python เป็นหนึ่งในภาษาคอมพิวเตอร์ที่สำคัญเป็นอันดับต้น ๆ ที่นักพัฒนาเลือกใช้ในการพัฒนาเทคโนโลยีด้วยเหตุผลหลากหลายประการไม่ว่าจะเป็นความง่ายในการเขียน การที่มี library รองรับการใช้งานที่หลากหลาย หรือแม้กระทั่งการที่มีชุมชนผู้ใช้งานขนาดใหญ่ จึงทำให้บริษัททั้งเล็กและใหญ่มีความต้องการบุคลากรที่เป็นนักพัฒนาภาษา Python เป็นจำนวนมาก แต่ผู้เชี่ยวชาญภาษา Python ก็ยังคงมีจำนวนไม่เพียงพอต่อความต้องการทำให้ค่าตัวของนักพัฒนาเหล่านี้พุ่งสูงขึ้นมาก

การขาดผู้เชี่ยวชาญภาษา Python ไม่ได้มีสาเหตุมาจากว่าไม่มีคนสนใจจะเข้ามาทำงานในสายนี้ อันที่จริงมีผู้คนจำนวนมากหลั่งไหลกันเข้ามาเพื่อที่จะเปลี่ยนสายการทำงานมาเป็นนักพัฒนา เนื่องจากค่าตอบแทนที่สูงจนน่าตกใจ แต่เหตุผลที่แท้จริงที่ทำให้ในอุตสาหกรรมยังคงขาดแคลนนักพัฒนา Python มาจากการที่ผู้เริ่มต้นส่วนใหญ่ไม่สามารถพัฒนาทักษะจนกลายเป็นผู้เชี่ยวชาญได้สำเร็จ นั่นจึงเป็นที่มาของหนังสือ ตะลุยโจทย์ Python 500 ข้อในมือทุกท่านเล่มนี้ โจทย์ทั้ง 500 ข้อถูกออกแบบมาให้ครอบคลุมหัวข้อที่สำคัญอย่างครบถ้วน และโจทย์จะมีการไล่ระดับเริ่มจากระดับง่ายก่อนเพื่อให้เกิดความคุ้นเคยกับโครงสร้างภาษา ไปจนถึงระดับยากเพื่อให้เกิดการประยุกต์และพัฒนาทักษะการเขียนโปรแกรมในระดับที่สูงขึ้นนั่นเอง

ที่มาของจำนวน 500 ข้อ มาจากแรงบันดาลใจจากอาจารย์ที่สอนพวกเราที่จุฬาลงกรณ์มหาวิทยาลัย อาจารย์เคยบอกว่า "การที่คนคนหนึ่งจะเก่งได้ ต้องมาจากการเก็บเที่ยวบินเยอะ ๆ พวกคุณยังไม่มีงานทำเพื่อเก็บเที่ยวบินก็ต้องใช้วิธีการทำโจทย์ปัญหา ถ้าคุณทำโจทย์ 50 ข้อ ฝีมือของคุณจะอยู่ในระดับพอใช้ได้ ถ้าคุณทำโจทย์ 200 ข้อ ฝีมือของคุณจะอยู่ในระดับเก่ง ถ้าคุณทำโจทย์ 500 ข้อ ฝีมือของคุณคือระดับเทพ" พวกเราที่ยึดถือแนวทางตามที่อาจารย์บอกก็ประสบความสำเร็จอย่างงดงามในการเรียนกันทุกคน เราจึงต้องการส่งต่อวิธีการฝึกฝนนี้เพื่อพัฒนาทักษะการเขียนโปรแกรมให้กับผู้อ่านทุกท่าน

สุดท้ายนี้ เราเชื่อว่าการที่ประเทศไทยมีนักพัฒนาภาษา Python เพิ่มขึ้นจะช่วยส่งเสริมการสรรค์สร้างนวัตกรรมใหม่ ๆ และพาประเทศไทยไปแข่งขันในเวทีระดับโลกได้

ทีมงาน TAUTOLOGY

สารบัญ

Page No.

0 How to use this book 1

วิธีการเข้าชมวิดีโอสอนเนื้อหา Python 1

วิธีการรับเอกสารประกอบการสอน 1

วิธีการรับ code เฉลย 2

วิธีการเข้าร่วม Facebook group 3

วิธีการดูตำแหน่งงาน Python ที่กำลังรับสมัคร 3

1 Introduction 4

ประวัติความเป็นมา Python 4

ข้อดีของ Python มีอะไรบ้าง 5

Python ถูกนำไปใช้ที่ไหนบ้าง 6

วิธีการติดตั้งใช้งาน Python 7

2 Print Command **12**

2.1 โจทย์ **12**

Print String 12

Print Numeric 14

Print String + Numeric 15

2.2 เฉลย **18**

Print String 18

Print Numeric 25

Print String + Numeric 27

3 Variable **33**

3.1 โจทย์ **33**

3.2 เฉลย 37

4 Data Structure 41

4.1 โจทย์ 41

Numeric 41

String 43

List 45

Tuple 48

Dictionary 49

Set 50

4.2 เฉลย 52

Numeric 52

String 58

List 65

Tuple 77

Dictionary 83

Set 88

5 Operator 94

5.1 โจทย์ 95

5.2 เฉลย 97

6 If-Else (Flowchart-Code) 100

6.1 โจทย์ 101

6.2 เฉลย 107

7 If-Else (Code-Flowchart) 124

7.1 โจทย์ 125

7.2 เฉลย 140

8 If-Else (Problem) 146
8.1 โจทย์ 146
8.2 เฉลย 155

9 For & While loop (Flowchart-Code) 190
9.1 โจทย์ 191
9.2 เฉลย 194

10 For & While loop (Code-Flowchart) 201
10.1 โจทย์ 202
10.2 เฉลย 208

11 For & While loop (Problem) 212
11.1 โจทย์ 212
11.2 เฉลย 214

12 Problem 224
12.1 โจทย์ 224
12.2 เฉลย 242

13 Function 353
13.1 โจทย์ 353
13.2 เฉลย 362

14 OOP 392
14.1 โจทย์ 392
14.2 เฉลย 404

How to use this book

• วิธีการเข้าชมวิดีโอสอนเนื้อหา Python

ในหน้าแรกของแต่ละบทจะมี QR code (ฝั่งซ้ายมือตามรูปข้างล่าง) สำหรับสแกนเพื่อเรียนวิดีโอเนื้อหา และเฉลยแบบฝึกหัดของแต่ละบท



• วิธีการรับเอกสารประกอบการสอน

1. กรอก google form สำหรับรับเอกสารประกอบการสอน และรับข่าวสารจากเรา ได้ที่ QR code ข้างล่าง



2. ระบบจะทำการส่ง e-mail ยืนยันสิทธิ์ให้ท่าน

3. เอกสารประกอบการสอนจะถูกแนบใน e-mail ยืนยันสิทธิ์ของท่าน

- วิธีการรับ code เฉลย

- การรับ code เฉลยบน google colab

ในหน้าแรกของแต่ละบทจะมี QR code (ฝั่งขวามือตามรูปข้างล่าง) สำหรับสแกนเพื่อรับเฉลยผ่าน google colab



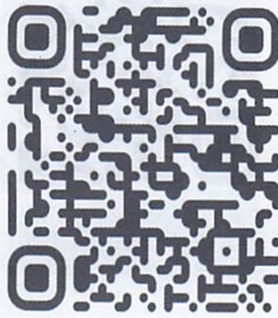
- การรับ code เฉลยบน Jupyter Notebook

หลังจากท่านได้รับการยืนยันสิทธิ์แล้ว ท่านจะได้รับ code เฉลยทาง e-mail

ถ้าหากท่านยังไม่ได้ยืนยันสิทธิ์ ให้ทำตามขั้นตอนที่ 1 และ 2 ของวิธีรับเอกสารประกอบการสอน

• วิธีการเข้าร่วม Facebook group

สแกน QR code เพื่อเข้าร่วม Facebook group ของหนังสือตะลุยโจทย์ Python 500 ข้อ เพื่อสอบถามข้อสงสัย แลกเปลี่ยนความคิดเห็น และอัพเดทข่าวสารต่าง ๆ ของหนังสือเล่มนี้



• วิธีการดูตำแหน่งงาน Python ที่กำลังสมัคร

สแกน QR code เพื่อดูตำแหน่งงานในสายงานภาษา Python ที่กำลังเปิดรับสมัคร โดยทางทีมงาน Tautology ได้รวบรวมตำแหน่งงานจากเว็บต่าง ๆ ไว้ในลิ้งนี้แบบ real time



หากท่านพบปัญหาในการลงทะเบียน หรือต้องการสอบถามข้อมูลเพิ่มเติม สามารถติดต่อได้ที่ Facebook : Tautology Thailand

01 INTRODUCTION

1



สแกนเพื่อเรียนเนื้อหาบทที่ 1

ประวัติความเป็นมาของ Python

ในช่วงปลายทศวรรษที่ 1980s Guido van Rossum นักพัฒนาที่ Centrum Wiskunde & Informatica (CWI) สถาบันวิจัยด้านคณิตศาสตร์และคอมพิวเตอร์ของประเทศเนเธอร์แลนด์ กำลังพัฒนาภาษา ABC ซึ่งเป็นภาษา high level ใช้งานง่าย แต่ข้อเสียคือ ติดต่อกับ server ระบบไฟล์ และระบบประมวลผลได้ยาก

ในช่วงวันหยุดยาวของสิ้นปี 1989 Guido ก็เกิดความคิดที่จะพัฒนาภาษาที่คล้ายคลึงกับภาษา ABC ขึ้นมา แต่จะต้องสามารถติดต่อกับระบบปฏิบัติการได้ดีด้วย รวมถึงพยายามดึงลักษณะเด่นของภาษาอื่น ๆ มาร่วมด้วย เกิดมาเป็นภาษา Python โดยชื่อ Python นั้นมาจาก รายการทีวี "Monty Python's Flying Circus" และหลังจากนั้น Guido ก็ใช้เวลาว่าง ในการพัฒนาภาษา Python ต่อไป

ต่อมาเพื่อนร่วมงานของ Guido ก็เริ่มได้ลองใช้ภาษา Python แล้วรู้สึกชอบจึงเข้ามาร่วมพัฒนาด้วย จนเริ่มแพร่หลายออกไปยังหน่วยงานอื่น ๆ ของ CWI และจากนั้นอีกหนึ่งปีจากการร่วมกันพัฒนา Python ของหลาย ๆ หน่วยงานใน CWI ทุกคนจึงตัดสินใจที่จะเปิดให้ Python เป็น open source ให้คนทั่วไปได้ใช้งานครั้งแรกในเดือนกุมภาพันธ์ปี 1991 และได้รับความนิยมอย่างล้นหลาม จนกลายมาเป็นหนึ่งในภาษาหลักที่ใช้พัฒนาโปรแกรมของทุกวันนี้

5

ข้อดีของ Python มีอะไรบ้าง ?

1. เป็นภาษาที่เขียนง่าย

Python มีโครงสร้างภาษาที่ง่าย เหมาะสำหรับผู้เริ่มต้น ไม่ซับซ้อนเท่ากับภาษาอื่น ๆ ทำให้ผู้ใช้สามารถทุ่มเทให้กับ logic การเขียนโปรแกรมได้เต็มที่

2. มี Community ของผู้ใช้งานขนาดใหญ่

Python มีจำนวนผู้ใช้งานสูงมากในปัจจุบัน ทำให้เมื่อเกิดปัญหาจากการใช้งานไม่ว่าจะเป็น bugs, error ต่าง ๆ หรือ การที่ผู้ใช้ไม่ทราบคำสั่ง ก็จะสามารถหาวิธีแก้ไขได้อย่างรวดเร็ว

3. มี Library ให้ใช้งานหลากหลาย

Python มี Library ให้เลือกใช้งานหลากหลาย และ ครอบคลุมการใช้งานในด้านต่าง ๆ อย่างครบครัน

4. เหมาะมาก ๆ กับงานทางด้าน Data

Python เป็นตัวเลือกอันดับหนึ่งในเวลานี้สำหรับงานด้าน Data โดยเฉพาะอย่างยิ่งงาน Data Science ที่มี Library ที่ Support มากมาย ไม่ว่าจะเป็น TensorFlow, Scikit-Learn, Keras หรือ PyTorch รวมถึงงานด้าน Data Visualization ที่ก็มี Library ที่ Support หลากหลาย เช่น Matplotlib, Seaborn และ Bokeh

5. มีความต้องการในตลาดแรงงานมาก

ตลาดแรงงานปัจจุบันมีความต้องการ Python Developer เป็นอย่างมาก อันเนื่องมาจากความนิยมของภาษาและความสามารถในการทำงานที่หลากหลาย บริษัทยักษ์ใหญ่ระดับโลกหลายแห่งก็มีความต้องการคนในตำแหน่งนี้ ไม่ว่าจะเป็น Google, J.P. Morgan, Bank of America, American Express หรือ Visa Inc.

(ref. https://www.facebook.com/tautologyai/posts/2987242824730932)

Python ถูกนำไปใช้ที่ไหนบ้าง ?

1 Python ได้เข้าไปอยู่ในชีวิตประจำวันของพวกเราอย่างมากมาย ไม่ว่าจะเป็นการสร้างเว็บ, การสร้างเกม, AI, Machine Learning รวมถึง App ต่าง ๆ ที่เราใช้กันในปัจจุบัน ไม่ว่าจะเป็น Instagram, Netflix, Spotify, Amazon

Netflix เป็นหนึ่งในบริษัทที่ทรงพลังทางด้านระบบการแนะนำหนังให้กับผู้ชม (Recommendation System) โดยระบบการแนะนำของ Netflix ได้เขียนด้วยภาษา Python

บริษัท E-Commerce ขนาดใหญ่ อย่างบริษัท Amazon ได้มีการเลือกใช้งานภาษา Python ในการจัดการระบบต่าง ๆ เช่น ระบบแนะนำสินค้า และระบบ Machine Learning เนื่องมาจากว่า Python เป็นหนึ่งใน Technology ที่ดีที่สุดในการจัดการข้อมูลขนาดใหญ่มาก ๆ

บริษัท Uber เป็นอีกหนึ่งบริษัทที่ต้องใช้คอมพิวเตอร์ในการช่วยคำนวณและทำนาย demand, supply ของลูกค้าที่เข้ามาใช้งาน รวมถึงเวลาของการเดินทางบนท้องถนน ซึ่งใช้พลังในการคำนวณค่อนข้างสูง ดังนั้นจึงต้องมีการเลือก platform ที่ใช้ในการคำนวณที่เหมาะสม ซึ่งตอนนั้นมีสองตัวเลือก นั้นก็คือ Ruby และ Python สุดท้ายแล้ว Uber ได้เลือกใช้ Python เป็น Platform หลักในการคำนวณ เพราะมีรูปแบบการใช้งานที่ง่าย และสามารถแก้ปัญหาใหญ่ ๆ ได้ดี

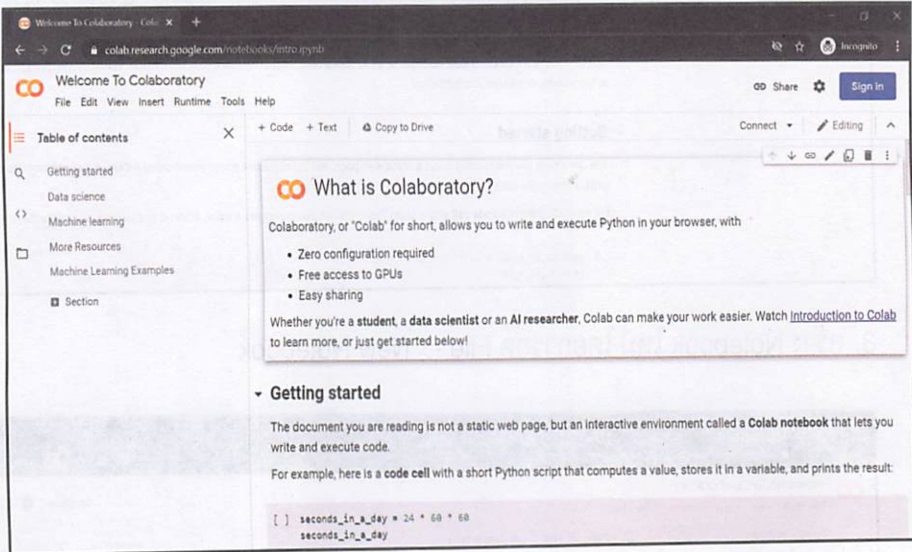
นอกจากนั้นแล้ว ในองค์กร หรือบริษัทต่าง ๆ ได้มีการนำ Python เข้ามาควบคุมระบบการจัดการต่าง ๆ เช่น การส่งอีเมล์ การค้นหาไฟล์งานในคอมพิวเตอร์ หรือการคำนวณต่าง ๆ เป็นต้น

(ref. https://www.edureka.co/blog/python-applications/)
(ref. https://www.netguru.com/blog/python-app-examples)

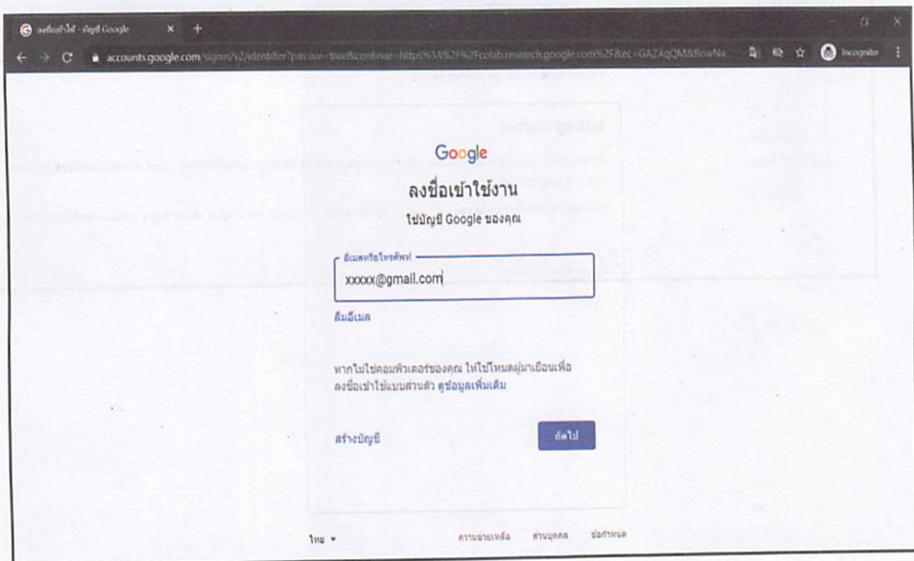
วิธีการติดตั้งใช้งาน Pyhton

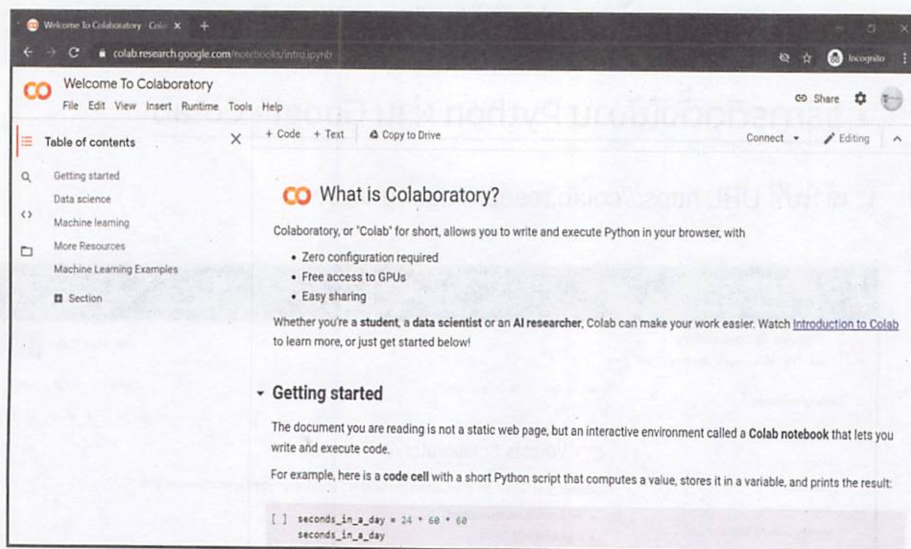
• วิธีการติดตั้งใช้งาน Python ผ่าน Google Colab

1. เข้าไปที่ URL https://colab.research.google.com/

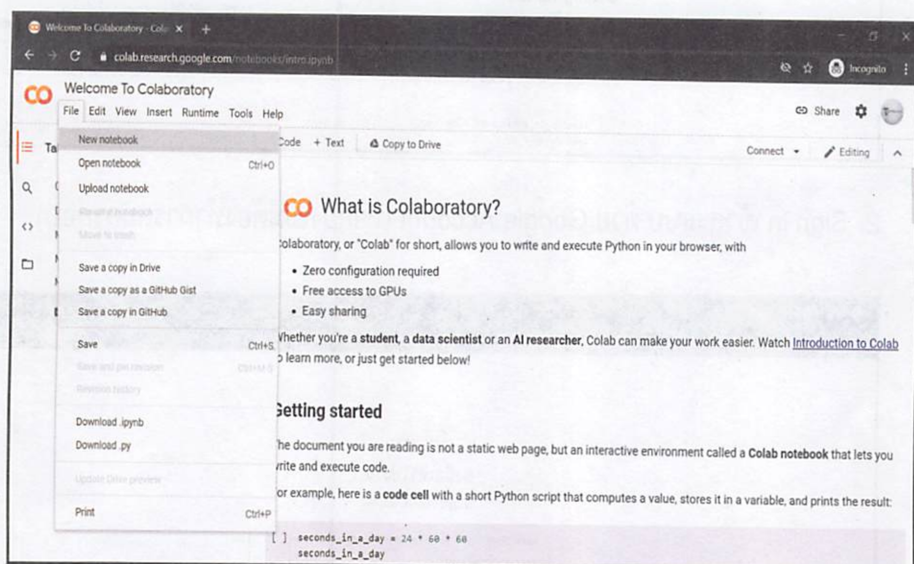


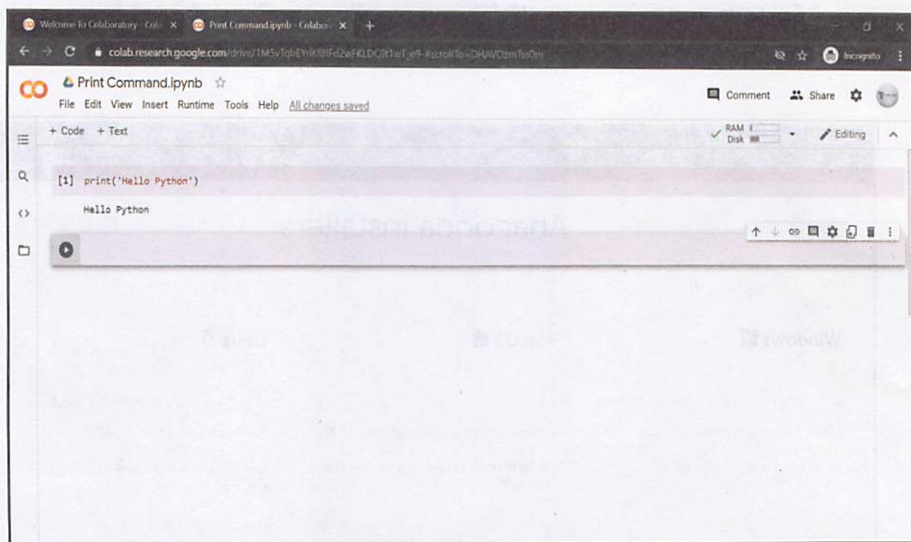
2. Sign in เข้าสู่ระบบ ด้วย Google Account (หากยังไม่มีต้องทำการสมัครก่อน)





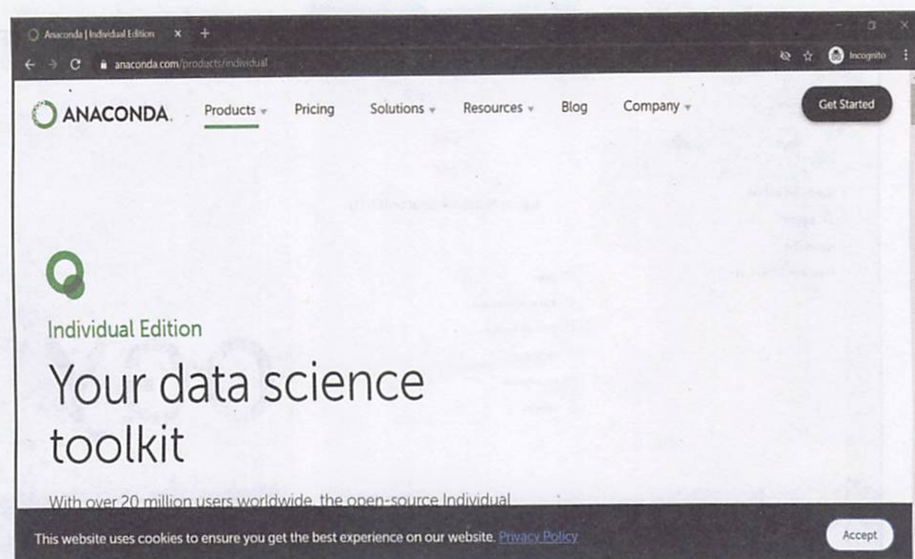
3. สร้าง Notebook ใหม่ โดยการกด File -> New Notebook





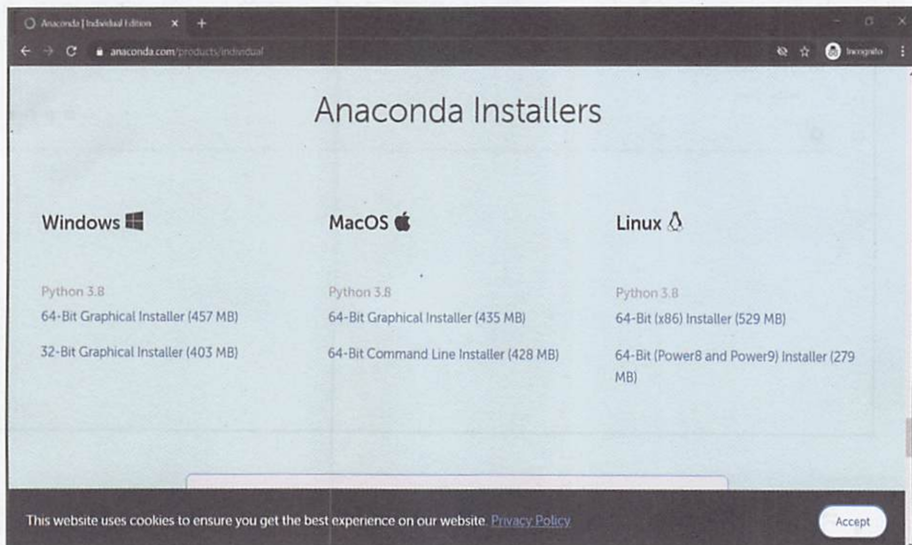
• วิธีการติดตั้ง Jupyter Notebook

1. ไปที่เว็บไซต์ https://www.anaconda.com/products/individual

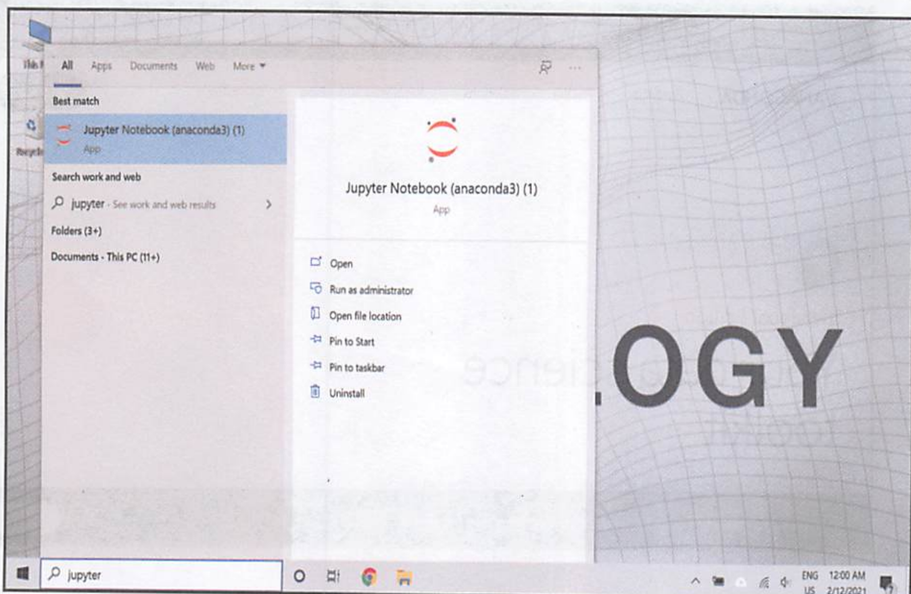


1

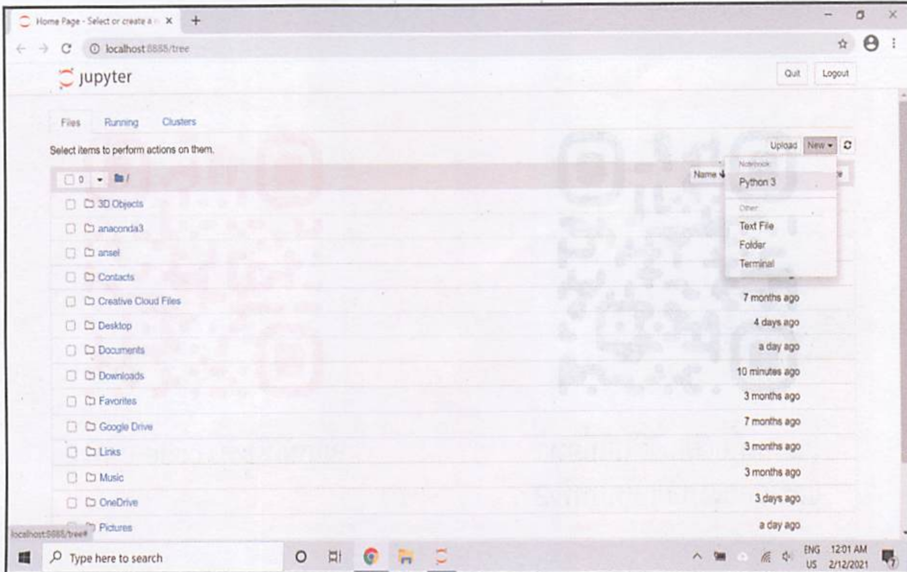
2. ดาวโหลด Python ตามระบบปฏิบัติการของเครื่องคอมพิวเตอร์

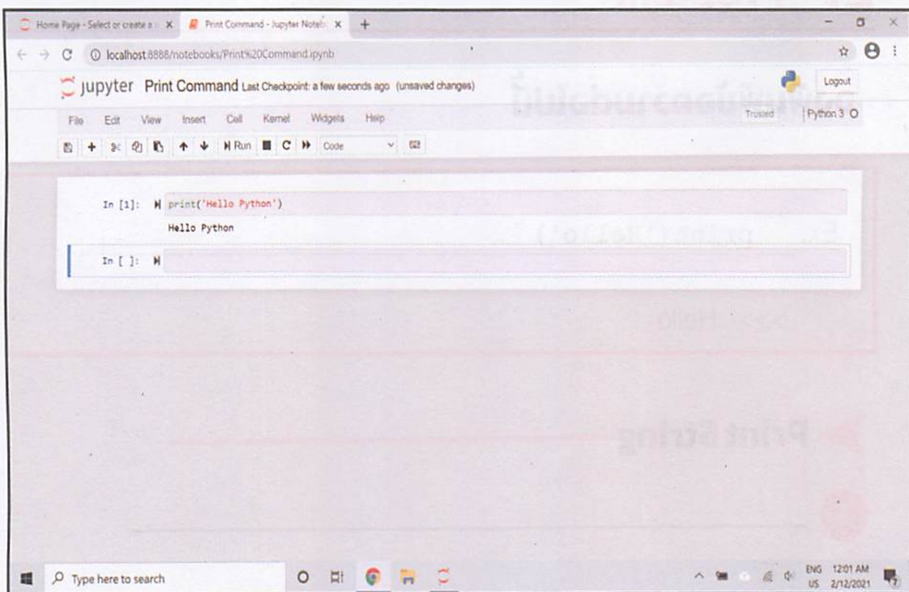


3. ติดตั้งโปรแกรมจนเสร็จสิ้น
4. เปิดเข้าไปที่ Jupyter Notebook ในเครื่องคอมพิวเตอร์



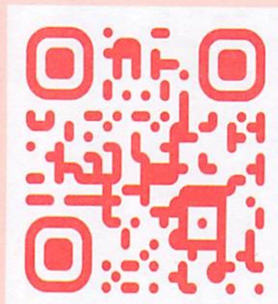
5. สร้าง Notebook ใหม่ โดยการกด New -> Python 3





02 PRINT COMMAND


สแกนเพื่อเรียนวิดีโอเนื้อหา
และเฉลยแบบฝึกหัดบทที่ 2


สแกนเพื่อรับ code บทที่ 2

2.1 Problem

จงพิมพ์ข้อความต่อไปนี้

Ex. `print('Hello')`

>>> Hello

Print String

1. ______________________

>>> Hello

2. ______________________

>>> Hello World

3. ______________________________

```
>>> hello_python
```

4. ______________________________

```
>>> Merry X'Mas
```

5. ______________________________

```
>>> I want to ask you "Why don't you drive to work ?"
```

6. ______________________________

```
>>> "I don't have a car"
```

7. ______________________________

```
>>> You got a new job !? That's so exciting
```

8. ______________________________

```
>>> สวัสดีวันจันทร์
```

9. ______________________________

```
>>> ความแตกต่างระหว่างคนเก่งกับคนไม่เก่ง คือ "การใช้เวลาว่าง
    ให้เป็นประโยชน์"
```

10. ______________________________

```
>>> /\/\/\
```

2

11. ______

```
>>> a
    an
    ant
```

12. ______

```
        *
*       *       *
        *
```

13. ______

```
*       +       *
+       *       +
*       +       *
```

14. ______

```
>>> Just because something
    thinks differently from you,
    does that mean it's not thinking ?
```

15. ______

```
\               /
        ×
/               \
```

Print Numeric

16. ______

```
>>> 25
```

17. ______________________________

>>> 100.000000

18. ______________________________

>>> 3.141592653589793

19. **a=2**

____

>>> 2

20. **a=12.5**

____

>>> 12.5

Print String + Numeric

21. **a = 2**
b = 3

____

>>> 2 x 3 = 6

22. **a = 2**
b = 3

____

>>> 2 + 3 = 3 + 2 = 5

2

23. a = 2
b = 3
c = 5

___

>>> 2*(3 + 5) = 2*3 + 2*5

24. a = 2.4
b = 2.5

___

>>> 2.4 + 2.5 = 4.9000

25. a = 5
b = 2

___

>>> 5.00 - 2.00 = 3.0000

26. birthday = 25

___

>>> ฉันเกิดวันที่ 25 ธันวาคม

27. a = 5
b = 100

___

>>> 5 เท่าของ 100 มีค่าเท่ากับ 500

28. **a = 3.5**

____

>>> เขามีเงินเยอะกว่าฉัน 3.50 บาท

29. **a = 5**

____

>>> ฉันได้กำไร 5 %

30. **a = 2**
b = 3.5

____

>>> เมื่อวานฉันขาดทุน 2 % วันนี้ฉันได้กำไร 3.50 %

2.2 Solution

Print String

2

เฉลยข้อที่ 1.

วิธีที่ 1. (recommend)

Input :

```
print('Hello')
```

Output :

```
Hello
```

วิธีที่ 2.

Input :

```
print("Hello")
```

Output :

```
Hello
```

เฉลยข้อที่ 2.

วิธีที่ 1. (recommend)

Input :

```
print('Hello World')
```

Output :

```
Hello World
```

วิธีที่ 2.

Input :

```
print("Hello World")
```

Output :

```
Hello World
```

เฉลยข้อที่ 3.

วิธีที่ 1. (recommend)

Input :

```
print('hello_python')
```

Output :

```
hello_python
```

วิธีที่ 2.

Input :

```
print("hello_python")
```

Output :

```
hello_python
```

เฉลยข้อที่ 4.

วิธีที่ 1. (recommend)

Input :

```
print("Merry X'mas")
```

Output :

```
Merry X’mas
```

วิธีที่ 2.

Input :

```
print('Merry X\'mas')
```

Output :

```
Merry X’mas
```

เฉลยข้อที่ 5.

วิธีที่ 1. (recommend)

Input :

```
print('I want to ask you "Why don\'t you
    ↪ drive to work ?"')
```

Output :

```
I want to ask you "Why don't you drive to
work ?"
```

วิธีที่ 2.

Input :

```
print("I want to ask you \"Why don't you
    ↪ drive to work ?\"")
```

Output :

```
I want to ask you "Why don't you drive to
work ?"
```

เฉลยข้อที่ 6.

วิธีที่ 1. (recommend)

Input :

```
print('"I don\'t have a car"')
```

Output :

```
"I don't have a car"
```

วิธีที่ 2.

Input :

```
print("\"I don't have a car\"")
```

Output :

```
"I don't have a car"
```

เฉลยข้อที่ 7.

วิธีที่ 1. (recommend)

Input :

```
print("You got a new job !? That's so
    ↪ exciting !")
```

Output :

```
You got a new job !? That's so exciting !
```

2

วิธีที่ 2.

Input :

```
print('You got a new job !? That\'s so
    ↪ exciting !')
```

Output :

```
You got a new job !? That's so exciting !
```

เฉลยข้อที่ 8.

วิธีที่ 1. (recommend)

Input :

```
print('สวัสดีวันจันทร์')
```

Output :

```
สวัสดีวันจันทร์
```

วิธีที่ 2.

Input :

```
print("สวัสดีวันจันทร์")
```

Output :

```
สวัสดีวันจันทร์
```

เฉลยข้อที่ 9.

วิธีที่ 1. (recommend)

Input :

```
print('ความแตกต่างระหว่างคนเก่งกับคนไม่เก่ง คือ "การใช้
    เวลาว่างให้เป็นประโยชน์"')
```

Output :

```
ความแตกต่างระหว่างคนเก่งกับคนไม่เก่ง คือ "การใช้เวลาว่างให้
เป็นประโยชน์"
```

วิธีที่ 2.

Input :

```
print("ความแตกต่างระหว่างคนเก่งกับคนไม่เก่ง คือ "การใช้
    เวลาว่างให้เป็นประโยชน์"")
```

Output :

ความแตกต่างระหว่างคนเก่งกับคนไม่เก่ง คือ "การใช้เวลาว่างให้เป็นประโยชน์"

เฉลยข้อที่ 10.

วิธีที่ 1. (recommend)

Input :

```
print('/\\/\\/\\')
```

Output :

```
/\/\/\
```

วิธีที่ 2.

Input :

```
print("/\\/\\/\\")
```

Output :

```
/\/\/\
```

เฉลยข้อที่ 11.

วิธีที่ 1. (recommend)

Input :

```
print('''a
          an
          ant''')
```

Output :

```
a
an
ant
```

วิธีที่ 2.

Input :

```
print("""a
          an
          ant""")
```

Output :

```
a
an
ant
```

เฉลยข้อที่ 12.

วิธีที่ 1. (recommend)

Input :

```
print('''\t*
          *\t*\t*
          \t*''')
```

Output :

```
        *
*       *       *
        *
```

วิธีที่ 2.

Input :

```
print("""\t*
          *\t*\t*
          \t*""")
```

Output :

```
        *
*       *       *
        *
```

เฉลยข้อที่ 13.

วิธีที่ 1. (recommend)

Input :

```
print('''*\t+\t*
          +\t*\t+
          *\t+\t*''')
```

Output :

```
*       +       *
+       *       +
*       +       *
```

2

วิธีที่ 2.

Input :

```
print("""*\t+\t*
         +\t*\t+
         *\t+\t*""")
```

Output :

```
*	+	*
+	*	+
*	+	*
```

เฉลยข้อที่ 14.

วิธีที่ 1. (recommend)

Input :

```
print("""Just because something
         thinks differently from you,
         does that mean it's not thinking ?
    ↪ """)
```

Output :

```
Just because something
thinks differently from you,
does that mean it's not thinking ?
```

วิธีที่ 2.

Input :

```
print('''Just because something
         thinks differently from you,
         does that mean it\'s not thinking ?
    ↪ ''')
```

Output :

```
Just because something
thinks differently from you,
does that mean it's not thinking ?
```

เฉลยข้อที่ 15.

วิธีที่ 1. (recommend)

Input :

```
print('''\\\t\t/
         \tX
         /\t\t\\''')
```

Output :

```
\               /
        X
/               \
```

วิธีที่ 2.

Input :

```
print("""\\\t\t/
         \tX
         /\t\t\\""")
```

Output :

```
\               /
        X
/               \
```

Print Numeric

เฉลยข้อที่ 16.

วิธีที่ 1. (recommend)

Input :

```
print(25)
```

Output :

```
25
```

วิธีที่ 2.

Input :

```
print('%d' %25)
```

Output :

```
25
```

2

เฉลยข้อที่ 17.

วิธีที่ 1. (recommend)

Input :

```
print('%f' %100)
```

Output :

```
100.000000
```

วิธีที่ 2.

Input :

```
print('%.6f' %100)
```

Output :

```
100.000000
```

เฉลยข้อที่ 18.

วิธีที่ 1. (recommend)

Input :

```
from math import pi
print('%.15f' %pi)
```

Output :

```
3.141592653589793
```

วิธีที่ 2.

Input :

```
import math
print('%.15f' %math.pi)
```

Output :

```
3.141592653589793
```

เฉลยข้อที่ 19.

วิธีที่ 1. (recommend)

Input :

```
a = 2
print(a)
```

Output :

```
2
```

วิธีที่ 2.

Input :

```
a = 2
print('%d' %a)
```

Output :

```
2
```

เฉลยข้อที่ 20.

วิธีที่ 1. (recommend)

Input :

```
a = 12.5
print(a)
```

Output :

```
12.5
```

วิธีที่ 2.

Input :

```
a = 12.5
print('%.1f' %a)
```

Output :

```
12.5
```

Print String + Numeric

เฉลยข้อที่ 21.

วิธีที่ 1. (recommend)

Input :

```
a = 2
b = 3
print(a, 'x', b, '=', a*b)
```

Output :

```
2 x 3 = 6
```

2

วิธีที่ 2.

Input :

```
a = 2
b = 3
print('%d x %d = %d' %(a, b, a*b))
```

Output :

```
2 x 3 = 6
```

เฉลยข้อที่ 22.

วิธีที่ 1. (recommend)

Input :

```
a = 2
b = 3
print('%d + %d = %d + %d = %d' %(a, b, b, a,
    ↪ a+b))
```

Output :

```
2 + 3 = 3 + 2 = 5
```

วิธีที่ 2.

Input :

```
a = 2
b = 3
print(a, '+', b, '=', b, '+', a, '=', a+b)
```

Output :

```
2 + 3 = 3 + 2 = 5
```

เฉลยข้อที่ 23.

วิธีที่ 1. (recommend)

Input :

```
a = 2
b = 3
c = 5
print('%d*(%d + %d) = %d*%d + %d*%d' %(a, b,
    ↪ c, a, b, a, c))
```

Output :

```
2*(3 + 5) = 2*3 + 2*5
```

วิธีที่ 2.

Input :

```
a = 2
b = 3
c = 5
print("%d*(%d + %d) = %d*%d + %d*%d" %(a, b,
    ↪ c, a, b, a, c))
```

Output :

```
2*(3 + 5) = 2*3 + 2*5
```

เฉลยข้อที่ 24.

วิธีที่ 1. (recommend)

Input :

```
a = 2.4
b = 2.5
print('%.1d + %.1d = %.4f' %(a, b, a+b))
```

Output :

```
2.4 + 2.5 = 4.9000
```

วิธีที่ 2.

Input :

```
a = 2.4
b = 2.5
print("%.1d + %.1d = %.4f" %(a, b, a+b))
```

Output :

```
2.4 + 2.5 = 4.9000
```

เฉลยข้อที่ 25.

วิธีที่ 1. (recommend)

Input :

```
a = 5
b = 2
print('%.1f - %.2f = %.4f' %(a, b, a-b))
```

Output :

```
5.0 - 2.00 = 3.0000
```

2

วิธีที่ 2.

Input :

```
a = 5
b = 2
print("%.1f - %.2f = %.4f" %(a, b, a-b))
```

Output :

```
5.0 - 2.00 = 3.0000
```

เฉลยข้อที่ 26.

วิธีที่ 1. (recommend)

Input :

```
birthday = 25
print('ฉันเกิดวันที่', birthday, 'ธันวาคม')
```

Output :

```
ฉันเกิดวันที่ 25 ธันวาคม
```

วิธีที่ 2.

Input :

```
birthday = 25
print("ฉันเกิดวันที่", birthday, "ธันวาคม")
```

Output :

```
ฉันเกิดวันที่ 25 ธันวาคม
```

เฉลยข้อที่ 27.

วิธีที่ 1. (recommend)

Input :

```
a = 5
b = 100
print('%d เท่าของ %d มีค่าเท่ากับ %d' %(a, b, a*b))
```

Output :

```
5 เท่าของ 100 มีค่าเท่ากับ 500
```

วิธีที่ 2.

Input :

```
a = 5
b = 100
print(a, 'เท่าของ', b, 'มีค่าเท่ากับ', a*b)
```

Output :

```
5 เท่าของ 100 มีค่าเท่ากับ 500
```


เฉลยข้อที่ 28.

วิธีที่ 1. (recommend)

Input :

```
a = 3.5
print('เขามีเงินเยอะกว่าฉัน %.2f บาท' %a)
```

Output :

```
เขามีเงินเยอะกว่าฉัน 3.50 บาท
```

วิธีที่ 2.

Input :

```
a = 3.5
print("เขามีเงินเยอะกว่าฉัน %.2f บาท" %a)
```

Output :

```
เขามีเงินเยอะกว่าฉัน 3.50 บาท
```

เฉลยข้อที่ 29.

วิธีที่ 1. (recommend)

Input :

```
a = 5
print('ฉันได้กำไร', a, '%')
```

Output :

```
ฉันได้กำไร 5 %
```

2

วิธีที่ 2.

Input :
```
a = 5
print('ฉันได้กำไร %d %%' %a)
```

Output :
```
ฉันได้กำไร 5 %
```

เฉลยข้อที่ 30.

วิธีที่ 1. (recommend)

Input :
```
a = 2
b = 3.5
print('เมื่อวานฉันขาดทุน %d %% วันนี้ฉันได้
    กำไร %.2f %%' %(a, b))
```

Output :
```
เมื่อวานฉันขาดทุน 2 % วันนี้ฉันได้กำไร 3.50 %
```

วิธีที่ 2.

Input :
```
a = 2
b = 3.5
print("เมื่อวานฉันขาดทุน %d %% วันนี้ฉันได้
    กำไร %.2f %%" %(a, b))
```

Output :
```
เมื่อวานฉันขาดทุน 2 % วันนี้ฉันได้กำไร 3.50 %
```

03 VARIABLE



สแกนเพื่อเรียนวิดีโอเนื้อหา
และเฉลยแบบฝึกหัดบทที่ 3



สแกนเพื่อรับ code บทที่ 3

3.1 problem

จงพิจารณาตัวแปรต่อไปนี้ว่าสามารถประกาศได้หรือไม่ ถ้าไม่ได้จงบอกเหตุผล

Ex. 5a = 20

___ ได้
_/_ ไม่ได้ เพราะ ตัวแปรไม่ขึ้นต้นด้วย **character** หรือ **underscore**

1. **name = 'ชื่อของคุณ'**

___ ได้
___ ไม่ได้ เพราะ ______________________________

2. **_name = 'ชื่อของคุณ'**

___ ได้
___ ไม่ได้ เพราะ ______________________________

3. name1 = 'ชื่อของคุณ'

___ ได้

___ ไม่ได้ เพราะ ______________________________

4. name! = 'ชื่อของคุณ'

___ ได้

___ ไม่ได้ เพราะ ______________________________

5. !name = 'ชื่อของคุณ'

___ ได้

___ ไม่ได้ เพราะ ______________________________

6. number = 20

___ ได้

___ ไม่ได้ เพราะ ______________________________

7. _number = 20

___ ได้

___ ไม่ได้ เพราะ ______________________________

8. @number = 20

___ ได้

___ ไม่ได้ เพราะ ______________________________

9. number@ = 20

___ ได้

___ ไม่ได้ เพราะ ______________________________

10. a5 = 20

___ ได้

___ ไม่ได้ เพราะ ______________________________

11. **5a = 20**

___ ได้

___ ไม่ได้ เพราะ ______________________________

12. **favorite_food = 'Hamburger'**

___ ได้

___ ไม่ได้ เพราะ ______________________________

13. **FavoriteFood = 'Hamburger'**

___ ได้

___ ไม่ได้ เพราะ ______________________________

14. **favorite food = 'Hamburger'**

___ ได้

___ ไม่ได้ เพราะ ______________________________

15. **if = 7**

___ ได้

___ ไม่ได้ เพราะ ______________________________

16. **else = 5**

___ ได้

___ ไม่ได้ เพราะ ______________________________

17. **for = 6**

___ ได้

___ ไม่ได้ เพราะ ______________________________

18. **a = 5, b = 6, c = 7**

___ ได้

___ ไม่ได้ เพราะ ______________________________

3

19. a, b, c = 5, 6, 7

___ ได้

___ ไม่ได้ เพราะ ______________________________

20. a = 5; b = 6; c = 7

___ ได้

___ ไม่ได้ เพราะ ______________________________

3.2 Solution

เฉลยข้อที่ 1.

ได้ เพราะ เป็นไปตามกฎการประกาศตัวแปรทั้ง 3 ข้อ อันได้แก่

- ขึ้นต้นด้วย character หรือ underscore
- ไม่มีอักขระพิเศษหรือเว้นวรรคในชื่อตัวแปร
- ไม่ใช้คำสงวนในภาษา Python

เฉลยข้อที่ 2.

ได้ เพราะ เป็นไปตามกฎการประกาศตัวแปรทั้ง 3 ข้อ อันได้แก่

- ขึ้นต้นด้วย character หรือ underscore
- ไม่มีอักขระพิเศษหรือเว้นวรรคในชื่อตัวแปร
- ไม่ใช้คำสงวนในภาษา Python

เฉลยข้อที่ 3.

ได้ เพราะ เป็นไปตามกฎการประกาศตัวแปรทั้ง 3 ข้อ อันได้แก่

- ขึ้นต้นด้วย character หรือ underscore
- ไม่มีอักขระพิเศษหรือเว้นวรรคในชื่อตัวแปร
- ไม่ใช้คำสงวนในภาษา Python

เฉลยข้อที่ 4.

ไม่ได้ เพราะ มีอักขระพิเศษ (!) ในชื่อตัวแปร

อักขระพิเศษในภาษา Python มีดังนี้ {, }, [,], !, @, #, $, %, ^, &, *, (,), -, =, /, |, \, ., <, >

เฉลยข้อที่ 5.

ไม่ได้ เพราะ ตัวแปรไม่ขึ้นต้นด้วย character หรือ underscore และ มีอักขระพิเศษ (!) ในชื่อตัวแปร

อักขระพิเศษในภาษา Python มีดังนี้ {, }, [,], !, @, #, $, %, ^, &, *, (,), -, =, /, |, \, ., <, >

เฉลยข้อที่ 6.

ได้ เพราะ เป็นไปตามกฎการประกาศตัวแปรทั้ง 3 ข้อ อันได้แก่

- ขึ้นต้นด้วย character หรือ underscore
- ไม่มีอักขระพิเศษหรือเว้นวรรคในชื่อตัวแปร
- ไม่ใช้คำสงวนในภาษา Python

เฉลยข้อที่ 7.

ได้ เพราะ เป็นไปตามกฎการประกาศตัวแปรทั้ง 3 ข้อ อันได้แก่

- ขึ้นต้นด้วย character หรือ underscore
- ไม่มีอักขระพิเศษหรือเว้นวรรคในชื่อตัวแปร
- ไม่ใช้คำสงวนในภาษา Python

เฉลยข้อที่ 8.

ไม่ได้ เพราะ ตัวแปรไม่ขึ้นต้นด้วย character หรือ underscore และ มีอักขระพิเศษ (@) ในชื่อตัวแปร

อักขระพิเศษในภาษา Python มีดังนี้ {, }, [,], !, @, #, $, %, ^, &, *, (,), -, =, /, |, \, ., <, >

เฉลยข้อที่ 9.

ไม่ได้ เพราะ มีอักขระพิเศษ (@) ในชื่อตัวแปร

อักขระพิเศษในภาษา Python มีดังนี้ {, }, [,], !, @, #, $, %, ^, &, *, (,), -, =, /, |, \, ., <, >

เฉลยข้อที่ 10.
ได้ เพราะ เป็นไปตามกฎการประกาศตัวแปรทั้ง 3 ข้อ อันได้แก่
- ขึ้นต้นด้วย character หรือ underscore
- ไม่มีอักขระพิเศษหรือเว้นวรรคในชื่อตัวแปร
- ไม่ใช้คำสงวนในภาษา Python

เฉลยข้อที่ 11.
ไม่ได้ เพราะ ตัวแปรไม่ขึ้นต้นด้วย character หรือ underscore

เฉลยข้อที่ 12.
ได้ เพราะ เป็นไปตามกฎการประกาศตัวแปรทั้ง 3 ข้อ อันได้แก่
- ขึ้นต้นด้วย character หรือ underscore
- ไม่มีอักขระพิเศษหรือเว้นวรรคในชื่อตัวแปร
- ไม่ใช้คำสงวนในภาษา Python

เฉลยข้อที่ 13.
ได้ เพราะ เป็นไปตามกฎการประกาศตัวแปรทั้ง 3 ข้อ อันได้แก่
- ขึ้นต้นด้วย character หรือ underscore
- ไม่มีอักขระพิเศษหรือเว้นวรรคในชื่อตัวแปร
- ไม่ใช้คำสงวนในภาษา Python

เฉลยข้อที่ 14.
ไม่ได้ เพราะ มีเว้นวรรคในชื่อตัวแปร

3

เฉลยข้อที่ 15.

ไม่ได้ เพราะ ใช้คำสงวนในภาษา Python

คำสงวนในภาษา Python มีดังนี้ and, assert, break, class, continue, def, del, elif, else, except, exec, finally, for, from, global, if, import, in, is, lambda, not, or, pass, print, raise, return, try, while, yield

เฉลยข้อที่ 16.

ไม่ได้ เพราะ ใช้คำสงวนในภาษา Python

คำสงวนในภาษา Python มีดังนี้ and, assert, break, class, continue, def, del, elif, else, except, exec, finally, for, from, global, if, import, in, is, lambda, not, or, pass, print, raise, return, try, while, yield

เฉลยข้อที่ 17.

ไม่ได้ เพราะ ใช้คำสงวนในภาษา Python

คำสงวนในภาษา Python มีดังนี้ and, assert, break, class, continue, def, del, elif, else, except, exec, finally, for, from, global, if, import, in, is, lambda, not, or, pass, print, raise, return, try, while, yield

เฉลยข้อที่ 18.

ไม่ได้ เพราะ ไม่เป็นไปตามกฎการประกาศตัวแปรหลายตัวพร้อมกัน

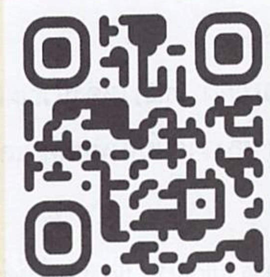
เฉลยข้อที่ 19.

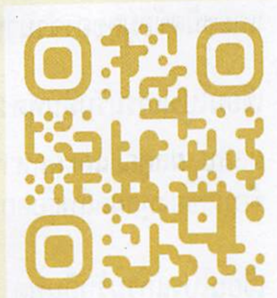
ได้ เพราะ เป็นไปตามกฎการประกาศตัวแปรหลายตัวพร้อมกัน

เฉลยข้อที่ 20.

ได้ เพราะ การใช้ ; เป็นการบอกขอบเขตคำสั่ง

04 DATA STRUCTURE


สแกนเพื่อเรียนวิดีโอเนื้อหา
และเฉลยแบบฝึกหัดบทที่ 4


สแกนเพื่อรับ code บทที่ 4

4.1 Problem

จงเขียน code ของโจทย์ต่อไปนี้

Numeric

1. เขียนโปรแกรมสร้างตัวแปรชื่อ numeric1 และกำหนดค่าให้เท่ากับ 5 หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ numeric1 ออกมา
2. เขียนโปรแกรมสร้างตัวแปรชื่อ numeric2 และกำหนดค่าให้เท่ากับ -3 หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ numeric2 ออกมา
3. เขียนโปรแกรมสร้างตัวแปรชื่อ numeric3 และกำหนดค่าให้เท่ากับ 0 หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ numeric3 ออกมา
4. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของจำนวนที่รับออกมา
5. เขียนโปรแกรมสร้างตัวแปรชื่อ numeric1 และกำหนดค่าให้เท่ากับ 5.0 หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ numeric1 ออกมา
6. เขียนโปรแกรมสร้างตัวแปรชื่อ numeric2 และกำหนดค่าให้เท่ากับ -3.1 หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ numeric2 ออกมา

7. เขียนโปรแกรมสร้างตัวแปรชื่อ numeric3 และกำหนดค่าให้เท่ากับ 0.0 หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ numeric3 ออกมา

8. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนจริง หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของจำนวนที่รับออกมา

9. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นจำนวนเต็ม (int1, int2) ให้คำนวณผลบวก, ผลลบ, ผลคูณ และผลหารของจำนวนที่รับมา หลังจากนั้นให้พิมพ์ค่าและชนิดข้อมูลของทุกผลลัพธ์ออกมา

10. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นจำนวนจริง (float1, float2) ให้คำนวณผลบวก, ผลลบ, ผลคูณ และผลหารของจำนวนที่รับมา หลังจากนั้นให้พิมพ์ค่าและชนิดข้อมูลของทุกผลลัพธ์ออกมา

11. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นจำนวนเต็มและจำนวนจริง (int1, float1) ให้คำนวณผลบวก, ผลลบ, ผลคูณ และผลหารของจำนวนที่รับมา หลังจากนั้นให้พิมพ์ค่าและชนิดข้อมูลของทุกผลลัพธ์ออกมา

12. เขียนโปรแกรมสร้างตัวแปรชื่อ logic1 และกำหนดค่าให้เท่ากับ True หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ logic1 ออกมา

13. เขียนโปรแกรมสร้างตัวแปรชื่อ logic2 และกำหนดค่าให้เท่ากับ False หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ logic2 ออกมา

14. กำหนดให้ logic1 = True และ logic2 = False เขียนโปรแกรมเพื่อนำค่าความจริงทั้งสองมาดำเนินการด้วยคำสั่ง “และ” หลังจากนั้นให้ทำการพิมพ์ค่าความจริงที่ได้จากการดำเนินการออกมา

15. กำหนดให้ logic1 = True และ logic2 = False เขียนโปรแกรมเพื่อนำค่าความจริงทั้งสองมาดำเนินการด้วยคำสั่ง “หรือ” หลังจากนั้นให้ทำการพิมพ์ค่าความจริงที่ได้จากการดำเนินการออกมา

16. เขียนโปรแกรมสร้างตัวแปรชื่อ complex1 และกำหนดค่าให้เท่ากับ 1 + 2j หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ complex1 ออกมา

17. เขียนโปรแกรมสร้างตัวแปรชื่อ complex1 และกำหนดค่าให้เท่ากับ 1 + 2j หลังจากนั้นให้ทำการพิมพ์ส่วนจริงออกมา

18. เขียนโปรแกรมสร้างตัวแปรชื่อ complex1 และกำหนดค่าให้เท่ากับ 1 + 2j หลังจากนั้นให้ทำการพิมพ์ส่วนจินตภาพออกมา

String

19. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ string1 ออกมา

20. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์อักขระที่ index เป็น 0 ออกมา

21. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์อักขระที่ index เป็น 1 ออกมา

22. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์อักขระที่ index เป็น -1 ออกมา

23. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์อักขระที่ index เป็น -2 ออกมา

24. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์อักขระ 't' ออกมา โดยอ้างอิงตำแหน่งจาก string1

25. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์อักขระ 'h' ออกมา โดยอ้างอิงตำแหน่งจาก string1

26. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์สายอักขระตั้งแต่ index ที่ 1 จนถึง index ที่ 3 ออกมา

27. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์สายอักขระตั้งแต่ index ที่ 2 จนถึง index ที่ -2 ออกมา

28. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์สายอักขระตั้งแต่ index ที่ 2 จนถึง index ที่ -1 ออกมา

4

29. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์สายอักขระตั้งแต่ index ที่ -4 จนถึง index ที่ 4 ออกมา

30. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์สายอักขระตั้งแต่ index ที่ -6 จนถึง index ที่ 4 ออกมา

4

31. เขียนโปรแกรมสร้างตัวแปรชื่อ string1 และกำหนดค่าให้เท่ากับ 'Python' หลังจากนั้นให้ทำการพิมพ์สายอักขระตั้งแต่ index ที่ -4 จนถึง index ที่ -2 ออกมา

32. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ จากนั้นให้ทำการพิมพ์ความยาวของสายอักขระออกมา

33. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ ให้ทำการตรวจสอบสายอักขระที่รับมาว่ามี 'ก' หรือไม่ จากนั้นให้พิมพ์ค่าความจริงออกมา

34. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นสายอักขระ (string1, string2) ให้ทำการตรวจ string1 ที่รับมาว่ามี string2 ในนั้นหรือไม่ จากนั้นให้พิมพ์ค่าความจริงออกมา

> **ตัวอย่างที่ 1 :** สมมติให้ string1 = 'happy', string2 = 'app' ผลลัพธ์ที่จะพิมพ์ออกมาคือ True

35. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ ให้ทำการแทนที่อักขระ 'a' ด้วย 'A' จากนั้นให้พิมพ์สายอักขระที่ผ่านการดำเนินการแล้วออกมา

36. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ ให้ทำการแทนที่อักขระ 'a' ด้วย 'A' และกำหนดค่าที่ผ่านการดำเนินการใส่ตัวแปรชื่อ string2 จากนั้นให้พิมพ์ string2 ออกมา

37. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ (sentence) ให้ทำการตัดคำในสายอักขระโดยใช้ ' ' เป็นตัวตัดคำ หลังจากนั้นให้พิมพ์ผลลัพธ์จากการตัดคำออกมา

> **ตัวอย่างที่ 1 :** สมมติให้ sentence = 'We love Python' ผลลัพธ์ที่จะพิมพ์ออกมาคือ ['We', 'love', 'Python']

38. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นสายอักขระ (sentence, c) ให้ทำการตัดคำใน sentence โดยใช้ c เป็นตัวตัดคำ หลังจากนั้นให้พิมพ์ผลลัพธ์จากการตัดคำออกมา

> **ตัวอย่างที่ 1 :** สมมติให้ sentence = 'Py-thon', c = '-' ผลลัพธ์ที่จะพิมพ์ออกมาคือ ['Py', 'thon']

39. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นสายอักขระ (str1, str2) ให้ทำการต่อสายอักขระทั้งสอง (โดยนำ str2 ไปต่อหลัง str1) หลังจากนั้นให้พิมพ์ผลลัพธ์ออกมา

> **ตัวอย่างที่ 1 :** สมมติให้ str1 = 'Fri', str2 = 'day' ผลลัพธ์ที่จะพิมพ์ออกมาคือ 'Friday'

List

40. เขียนโปรแกรมสร้าง list ที่มีสมาชิกเป็น 0, 1, 2, 'a', 'b', 'c' ตามลำดับ จากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ list ที่สร้างออกมา

41. กำหนดให้ list1 = [0, 1, 2, 'a', 'b', 'c'] เขียนโปรแกรมเพื่อพิมพ์สมาชิกที่ index เป็น 2 ออกมา

42. กำหนดให้ list1 = [0, 1, 2, 'a', 'b', 'c'] เขียนโปรแกรมเพื่อพิมพ์สมาชิกที่ index เป็น -3 ออกมา

43. กำหนดให้ list1 = [0, 1, 2, 'a', 'b', 'c'] เขียนโปรแกรมเพื่อพิมพ์ 0 ออกมาโดยอ้างอิงตำแหน่งจาก list1

44. กำหนดให้ list1 = [0, 1, 2, 'a', 'b', 'c'] เขียนโปรแกรมเพื่อพิมพ์ 'b' ออกมาโดยอ้างอิงตำแหน่งจาก list1

45. กำหนดให้ list1 = [0, 1, 2, 'a', 'b', 'c'] เขียนโปรแกรมเพื่อพิมพ์ list ที่มีสมาชิกตั้งแต่ index ที่ 2 จนถึง index ที่ 3 ของ list1 ออกมา

46. กำหนดให้ list1 = [0, 1, 2, 'a', 'b', 'c'] เขียนโปรแกรมเพื่อพิมพ์ list ที่มีสมาชิกตั้งแต่ index ที่ 1 จนถึง index ที่ -2 ของ list1 ออกมา

4

47. กำหนดให้ list1 = [0, 1, 2, 'a', 'b', 'c'] เขียนโปรแกรมเพื่อพิมพ์ [1, 2, 'a', 'b'] โดยอ้างอิงตำแหน่งจาก list1

48. กำหนดให้ list1 = [0, 1, 2, 'a', 'b', 'c'] เขียนโปรแกรมเพื่อพิมพ์ ['a', 'b', 'c'] โดยอ้างอิงตำแหน่งจาก list1

49. กำหนดให้ list1 = [0, 1, 2, 'a', 'b', 'c'] เขียนโปรแกรมเพื่อพิมพ์ [0, 1, 2] โดยอ้างอิงตำแหน่งจาก list1

50. กำหนดให้ list2 = ['ant', 'bird', 'cat', 'dog', 'eagle'] เขียนโปรแกรมเพื่อเปลี่ยนค่าสมาชิกใน index ที่ 3 เป็น 'duck' หลังจากนั้นให้พิมพ์ list2 ออกมา

51. กำหนดให้ list2 = ['ant', 'bird', 'cat', 'dog', 'eagle'] เขียนโปรแกรมเพื่อเปลี่ยน 'ant' เป็น 'ape' หลังจากนั้นให้พิมพ์ list2 ออกมา

52. กำหนดให้ list2 = ['ant', 'bird', 'cat', 'dog', 'eagle'] เขียนโปรแกรมเพื่อเพิ่ม 'fish' ไปเป็นสมาชิกตัวสุดท้ายของ list2 หลังจากนั้นให้พิมพ์ list2 ออกมา

53. เขียนโปรแกรมสร้าง empty list และรับอินพุต 3 ตัวที่เป็นสายอักขระ (str1, str2, str3) จากนั้นให้ทำการเพิ่มสายอักขระทั้ง 3 ไปเป็นสมาชิกใน list โดยใช้เมธอด append สุดท้ายให้พิมพ์ list ออกมา

54. กำหนดให้ list3 = ['apple', 'cherry', 'eggfruit'] เขียนโปรแกรมเพื่อแทรก 'banana' ไปเป็นสมาชิกใน index ที่ 1 ของ list3 หลังจากนั้นให้พิมพ์ list3 ออกมา

55. กำหนดให้ list3 = ['apple', 'banana', 'cherry', 'eggfruit'] เขียนโปรแกรมเพื่อแทรก 'kiwi' ไปเป็นสมาชิกใน index ที่ -2 ของ list3 หลังจากนั้นให้พิมพ์ list3 ออกมา

56. กำหนดให้ list4 = [0, 4, 2, 3, 1] เขียนโปรแกรมเพื่อเรียงลำดับข้อมูลใน list4 จากน้อยไปมาก หลังจากนั้นให้พิมพ์ list4 ออกมา

57. กำหนดให้ list4 = [0, 4, 2, 3, 1] เขียนโปรแกรมเพื่อเรียงลำดับข้อมูลใน list4 จากมากไปน้อย หลังจากนั้นให้พิมพ์ list4 ออกมา

58. กำหนดให้ list4 = [0, 4, 2, 3, 1] เขียนโปรแกรมเพื่อเรียงลำดับข้อมูลใน list4 จากน้อยไปมาก และเก็บค่าไว้ในตัวแปรชื่อ sorted_list4 หลังจากนั้นให้พิมพ์ sorted_list4 ออกมา

59. กำหนดให้ list4 = [0, 4, 2, 3, 1] เขียนโปรแกรมเพื่อเรียงลำดับข้อมูลใน list4 จากมากไปน้อย และเก็บค่าไว้ในตัวแปรชื่อ sorted_list4 หลังจากนั้นให้พิมพ์ sorted_list4 ออกมา

60. กำหนดให้ list5 = ['d', 'a', 'c', 'b', 'e'] เขียนโปรแกรมเพื่อเรียงลำดับข้อมูลใน list5 ตามตัวอักษร หลังจากนั้นให้พิมพ์ list5 ออกมา

61. กำหนดให้ list5 = ['d', 'a', 'c', 'b', 'e'] เขียนโปรแกรมเพื่อเรียงลำดับข้อมูลใน list5 ตามตัวอักษรแบบถอยหลัง และเก็บค่าไว้ในตัวแปรชื่อ sorted_list5 หลังจากนั้นให้พิมพ์ sorted_list5 ออกมา

62. กำหนดให้ list1 = ['a', 'b', 'c'] และ list2 = [0, 1, 2] เขียนโปรแกรมเพื่อนำสมาชิกทั้งหมดของ list2 ไปต่อท้าย list1 จากนั้นให้พิมพ์ list1 และ list2 ออกมา

63. กำหนดให้ list1 = ['a', 'b', 'c'] และ list2 = [0, 1, 2] เขียนโปรแกรมเพื่อนำสมาชิกทั้งหมดของ list2 ไปต่อท้าย list1 และเก็บ list ที่ได้จากการดำเนินการไว้ในตัวแปรชื่อ list3 จากนั้นให้พิมพ์ list1, list2 และ list3 ออกมา

64. กำหนดให้ list1 = [1, 2, 3, 1, 2, 3] เขียนโปรแกรมลบสมาชิกใน index ที่ 1 ออกจาก list1 โดยใช้คำสั่ง del หลังจากนั้นให้พิมพ์ list1 ออกมา

65. กำหนดให้ list1 = [1, 2, 3, 1, 2, 3] เขียนโปรแกรมลบ 3 ทั้งหมดออกจาก list1 โดยใช้คำสั่ง del หลังจากนั้นให้พิมพ์ list1 ออกมา

66. กำหนดให้ list1 = [1, 2, 3, 1, 2, 3] เขียนโปรแกรมลบ 1 ออกจาก list1 โดยใช้เมธอด remove หลังจากนั้นให้พิมพ์ list1 ออกมา

67. กำหนดให้ list1 = [1, 2, 3, 1, 2, 3] เขียนโปรแกรมลบ 1 ทั้งหมดออกจาก list1 โดยใช้เมธอด remove หลังจากนั้นให้พิมพ์ list1 ออกมา

68. กำหนดให้ list1 = [1, 2, 3, 1, 2, 3] เขียนโปรแกรมลบสมาชิกทุกตัวออกจาก list1 หลังจากนั้นให้พิมพ์ list1 ออกมา

69. กำหนดให้ list1 = [1, 2, 3, 'a', 'b', 'c'] เขียนโปรแกรมหาความยาวของ list1 และพิมพ์ออกมา

70. กำหนดให้ list1 = [1, 2, 3, 4, 5] เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม (numeric1) ให้ทำการตรวจว่า numeric1 เป็นสมาชิกใน list1 หรือไม่ จากนั้นให้พิมพ์ค่าความจริงออกมา

Tuple

71. เขียนโปรแกรมสร้าง tuple ที่มีสมาชิกเป็น 'America', 'Brazil', 'China', 'Dominican', 'Egypt' ตามลำดับ จากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ tuple ที่สร้างออกมา

72. กำหนดให้ tuple1 = ('America', 'Brazil', 'China', 'Dominican', 'Egypt') เขียนโปรแกรมเพื่อพิมพ์สมาชิกที่ index เป็น 1 ออกมา

73. กำหนดให้ tuple1 = ('America', 'Brazil', 'China', 'Dominican', 'Egypt') เขียนโปรแกรมเพื่อพิมพ์สมาชิกที่ index เป็น -2 ออกมา

74. กำหนดให้ tuple1 = ('America', 'Brazil', 'China', 'Dominican', 'Egypt') เขียนโปรแกรมเพื่อพิมพ์ 'China' ออกมาโดยอ้างอิงตำแหน่งจาก tuple1

75. กำหนดให้ tuple1 = ('America', 'Brazil', 'China', 'Dominican', 'Egypt') เขียนโปรแกรมเพื่อพิมพ์ 'Egypt' ออกมาโดยอ้างอิงตำแหน่งจาก tuple1

76. กำหนดให้ tuple2 = ('one', 'two', 'three', 1, 2, 3) เขียนโปรแกรมเพื่อพิมพ์ tuple ที่มีสมาชิกตั้งแต่ index ที่ 1 จนถึง index ที่ 3 ของ tuple2 ออกมา

77. กำหนดให้ tuple2 = ('one', 'two', 'three', 1, 2, 3) เขียนโปรแกรมเพื่อพิมพ์ tuple ที่มีสมาชิกตั้งแต่ index ที่ 2 จนถึง index ที่ -2 ของ tuple2 ออกมา

78. กำหนดให้ tuple2 = ('one', 'two', 'three', 1, 2, 3) เขียนโปรแกรมเพื่อพิมพ์ ('three', 1) โดยอ้างอิงตำแหน่งจาก tuple2

79. กำหนดให้ tuple2 = ('one', 'two', 'three', 1, 2, 3) เขียนโปรแกรมเพื่อพิมพ์ ('one', 'two', 'three') โดยอ้างอิงตำแหน่งจาก tuple2

80. กำหนดให้ tuple2 = ('one', 'two', 'three', 1, 2, 3) เขียนโปรแกรมเพื่อพิมพ์ (1, 2, 3) โดยอ้างอิงตำแหน่งจาก tuple2

81. กำหนดให้ tuple3 = ('one', 'two', 'three', 'four', 'five') เขียนโปรแกรมหาความยาวของ tuple3 และพิมพ์ออกมา

82. กำหนดให้ tuple3 = ('one', 'two', 'three', 'four', 'five') เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ (str1) ให้ทำการตรวจว่า str1 เป็นสมาชิกใน tuple3 หรือไม่ จากนั้นให้พิมพ์ค่าความจริงออกมา

Dictionary

83. เขียนโปรแกรมสร้าง dictionary ที่มี key เป็น 'first_name' และ 'last_name' โดยมี value เป็น 'John' และ 'Doe' ตามลำดับ จากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ dictionary ที่สร้างออกมา

84. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe'} เขียนโปรแกรมพิมพ์ value ของ key 'first_name' ออกมา

85. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe'} เขียนโปรแกรมพิมพ์ value ของ key 'last_name' ออกมา

86. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe'} เขียนโปรแกรมพิมพ์ list ของ key ทั้งหมดใน dict1 ออกมา

87. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe'} เขียนโปรแกรมพิมพ์ list ของ value ทั้งหมดใน dict1 ออกมา

88. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe'} เขียนโปรแกรมเปลี่ยน value ของ key 'first_name' เป็น 'Jane' หลังจากนั้นให้พิมพ์ dict1 ออกมา

89. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe'} เขียนโปรแกรมสร้าง key ใหม่ชื่อ 'age' และกำหนด value เป็น 32 หลังจากนั้นให้พิมพ์ dict1 ออกมา

90. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe'} เขียนโปรแกรมสร้าง key ใหม่ชื่อ 'age' และ 'hobby' โดยมี value เป็น 32 และ ['coding', 'studying'] ตามลำดับ หลังจากนั้นให้พิมพ์ dict1 ออกมา

91. เขียนโปรแกรมสร้าง empty dictionary และรับอินพุต 2 ตัวที่เป็นสายอักขระ (str1, str2) โดยให้ str1 เป็น key และ str2 เป็น value หลังจากนั้นให้พิมพ์ dictionary ออกมา

4

4

92. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe', 'age' : 32} เขียนโปรแกรมลบ key 'age' ออกจาก dict1 หลังจากนั้นให้พิมพ์ dict1 ออกมา

93. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe', 'age' : 32} เขียนโปรแกรมลบข้อมูลทั้งหมดออกจาก dict1 หลังจากนั้นให้พิมพ์ dict1 ออกมา

94. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe', 'age' : 32} เขียนโปรแกรมพิมพ์จำนวน key ของ dict1 ออกมา

95. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe', 'age' : 32} เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ (str1) ให้ทำการตรวจว่า str1 เป็นหนึ่งใน key ของ dict1 หรือไม่ จากนั้นให้พิมพ์ค่าความจริงออกมา

96. กำหนดให้ dict1 = {'first_name' : 'John', 'last_name' : 'Doe', 'age' : 32} เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ (str1) ให้ทำการตรวจว่า str1 เป็นหนึ่งใน value ของ dict1 หรือไม่ จากนั้นให้พิมพ์ค่าความจริงออกมา

Set

97. เขียนโปรแกรมสร้าง set ที่มีสมาชิกเป็น 1, 2, 3, 'i', 'ii', 'iii' จากนั้นให้ทำการพิมพ์ค่าและชนิดข้อมูลของ set ที่สร้างออกมา

98. กำหนดให้ set1 = {1, 2, 3, 'i', 'ii', 'iii'} เขียนโปรแกรมพิมพ์สมาชิกทุกตัวใน set1 ออกมา

99. กำหนดให้ set1 = {1, 2, 3, 'i', 'ii', 'iii'} เขียนโปรแกรมเพิ่ม 'iv' ไปเป็นสมาชิกใน set1 จากนั้นให้พิมพ์ set1 ออกมา

100. กำหนดให้ set1 = {1, 2, 3, 'i', 'ii', 'iii'} เขียนโปรแกรมเพิ่ม 1 ไปเป็นสมาชิกใน set1 จากนั้นให้พิมพ์ set1 ออกมา

101. เขียนโปรแกรมสร้าง empty set และรับอินพุต 3 ตัวที่เป็นสายอักขระ (str1, str2, str3) จากนั้นให้ทำการเพิ่มสายอักขระทั้ง 3 ไปเป็นสมาชิกใน set สุดท้ายให้พิมพ์ set ออกมา

102. เขียนโปรแกรมสร้าง empty set และรับอินพุต 3 ตัวที่เป็นสายอักขระ (str1, str2, str3) จากนั้นให้ทำการเพิ่มสายอักขระทั้ง 3 ไปเป็นสมาชิกใน set โดยการใช้เมธอด update สุดท้ายให้พิมพ์ set ออกมา

103. กำหนดให้ set1 = {1, 2, 3, 'i', 'ii', 'iii'} เขียนโปรแกรมลบ 'i' จาก set1 จากนั้นให้พิมพ์ set1 ออกมา

104. กำหนดให้ set1 = {1, 2, 3, 'i', 'ii', 'iii'} เขียนโปรแกรมพิมพ์จำนวนสมาชิกใน set1 ออกมา

105. กำหนดให้ set1 = {1, 2, 3, 'i', 'ii', 'iii'} เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม (int1) ให้ทำการตรวจว่า int1 เป็นสมาชิกของ set1 หรือไม่ จากนั้นให้พิมพ์ค่าความจริงออกมา

106. กำหนดให้ set1 = {1, 2, 3, 4, 5} และ set2 = {3, 4, 5, 6} เขียนโปรแกรมพิมพ์ผลลัพธ์ที่เกิดจากการ union ระหว่าง set1 และ set2 ออกมา

107. กำหนดให้ set1 = {1, 2, 3, 4, 5} และ set2 = {3, 4, 5, 6} เขียนโปรแกรมพิมพ์ผลลัพธ์ที่เกิดจากการ intersection ระหว่าง set1 และ set2 ออกมา

108. กำหนดให้ set1 = {1, 2, 3, 4, 5} และ set2 = {3, 4, 5, 6} เขียนโปรแกรมพิมพ์ผลลัพธ์ที่เกิดจาก set1 - set2 ออกมา

109. กำหนดให้ set1 = {1, 2, 3, 4, 5} และ set2 = {3, 4, 5, 6} เขียนโปรแกรมพิมพ์ผลลัพธ์ที่เกิดจาก set2 - set1 ออกมา

110. กำหนดให้ set1 = {1, 2, 3, 4, 5} และ set2 = {3, 4, 5, 6} เขียนโปรแกรมพิมพ์ผลลัพธ์ที่เกิดจากการ symmetric difference ระหว่าง set1 และ set2 ออกมา

4.2 Solution

Numeric

เฉลยข้อที่ 1.

Input :

```
numeric1 = 5
print(numeric1)
print(type(numeric1))
```

Output :

```
5
<class 'int'>
```

เฉลยข้อที่ 2.

Input :

```
numeric2 = -3
print(numeric2)
print(type(numeric2))
```

Output :

```
-3
<class 'int'>
```

เฉลยข้อที่ 3.

Input :

```
numeric3 = 0
print(numeric3)
print(type(numeric3))
```

Output :

```
0
<class 'int'>
```

เฉลยข้อที่ 4.

Input :

```
numeric4 = int(input('Please insert integer: '))
print(numeric4)
print(type(numeric4))
```

Output :

```
Please insert integer: 3
3
<class 'int'>
```

เฉลยข้อที่ 5.

Input :

```
numeric1 = 5.0
print(numeric1)
print(type(numeric1))
```

Output :

```
5.0
<class 'float'>
```


เฉลยข้อที่ 6.

Input :

```
numeric2 = -3.1
print(numeric2)
print(type(numeric2))
```

Output :

```
-3.1
<class 'float'>
```

เฉลยข้อที่ 7.

Input :

```
numeric3 = 0.0
print(numeric3)
print(type(numeric3))
```

Output :

```
0.0
<class 'float'>
```

เฉลยข้อที่ 8.

Input :

```
numeric4 = float(input('Please insert floating
    ↪ number: '))
print(numeric4)
print(type(numeric4))
```

Output :

```
Please insert floating number: 3.99
3.99
<class 'float'>
```

เฉลยข้อที่ 9.

Input :

```
int1 = int(input('Please insert int1: '))
int2 = int(input('Please insert int2: '))
print('int1:', int1, type(int1))

print('int2:', int2, type(int2))

a = int1 + int2
print('ผลบวก:', a, type(a))

b = int1 - int2
print('ผลลบ:', b, type(b))

c = int1*int2
print('ผลคูณ:', c, type(c))

d = int1/int2
print('ผลหาร:', d, type(d))
```

Output :

```
Please insert int1: 17
Please insert int2: 4
int1: 17 <class 'int'>
int2: 4 <class 'int'>
ผลบวก: 21 <class 'int'>
ผลลบ: 13 <class 'int'>
ผลคูณ: 68 <class 'int'>
ผลหาร: 4.25 <class 'float'>
```

เฉลยข้อที่ 10.

Input :

```
float1 = float(input('Please insert float1: '))
float2 = float(input('Please insert float2: '))
print('float1:', float1, type(float1))

print('float2:', float2, type(float2))

a = float1 + float2
print('ผลบวก:', a, type(a))

b = float1 - float2
print('ผลลบ:', b, type(b))

c = float1*float2
print('ผลคูณ:', c, type(c))

d = float1/float2
print('ผลหาร:', d, type(d))
```

Output :

```
Please insert float1: 24.5
Please insert float2: 2.4
float1: 24.5 <class 'float'>
float2: 2.4 <class 'float'>
ผลบวก: 26.9 <class 'float'>
ผลลบ: 22.1 <class 'float'>
ผลคูณ: 58.8 <class 'float'>
ผลหาร: 10.208333333333334 <class 'float'>
```

เฉลยข้อที่ 11.

Input :

```
int1 = int(input('Please insert int1: '))
float1 = float(input('Please insert float1: '))
print('int1:', int1, type(int1))

print('float1:', float1, type(float1))

a = int1 + float1
print('ผลบวก:', a, type(a))

b = int1 - float1
print('ผลลบ:', b, type(b))

c = int1*float1
print('ผลคูณ:', c, type(c))

d = int1/float1
print('ผลหาร:', d, type(d))
```

Output :

```
Please insert int1: 5
Please insert float1: 7.1
int1: 5 <class 'int'>
float1: 7.1 <class 'float'>
ผลบวก: 12.1 <class 'float'>
ผลลบ: -2.0999999999999996 <class 'float'>
ผลคูณ: 35.5 <class 'float'>
ผลหาร: 0.7042253521126761 <class 'float'>
```

เฉลยข้อที่ 12.

Input :

```
logic1 = True
print(logic1)
print(type(logic1))
```

Output :

```
True
<class 'bool'>
```

เฉลยข้อที่ 13.

Input :

```
logic2 = False
print(logic2)
print(type(logic2))
```

Output :

```
False
<class 'bool'>
```

4

เฉลยข้อที่ 14.

Input :

```
logic1 = True
logic2 = False
print((logic1 and logic2))
```

Output :

```
False
```

เฉลยข้อที่ 15.

Input :

```
logic1 = True
logic2 = False
print((logic1 or logic2))
```

Output :

```
True
```

เฉลยข้อที่ 16.

Input :

```
complex1 = 1 + 2j
print(complex1)
print(type(complex1))
```

Output :

```
(1+2j)
<class 'complex'>
```

เฉลยข้อที่ 17.

Input :

```
complex1 = 1 + 2j
print(complex1.real)
```

Output :

```
1.0
```

4

เฉลยข้อที่ 18.

Input :

```
complex1 = 1 + 2j
print(complex1.imag)
```

Output :

```
2.0
```

String

เฉลยข้อที่ 19.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
string1 = 'Python'
print(string1)
print(type(string1))
```

Output :

```
Python
<class 'str'>
```

เฉลยข้อที่ 20.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
string1 = 'Python'
print(string1[0])
```

Output :

```
P
```

เฉลยข้อที่ 21.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
string1 = 'Python'
print(string1[1])
```

Output :

```
y
```

เฉลยข้อที่ 22.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
string1 = 'Python'
print(string1[-1])
```

Output :

```
n
```

เฉลยข้อที่ 23.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
string1 = 'Python'
print(string1[-2])
```

Output :

```
o
```

เฉลยข้อที่ 24.

4

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
string1 = 'Python'
print(string1[2])
```

Output :

```
t
```

วิธีที่ 2.

Input :

```
string1 = 'Python'
print(string1[-4])
```

Output :

```
t
```

เฉลยข้อที่ 25.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
string1 = 'Python'
print(string1[3])
```

Output :

```
h
```

วิธีที่ 2.

Input :

```
string1 = 'Python'
print(string1[-3])
```

Output :

```
h
```


เฉลยข้อที่ 26.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
string1 = 'Python'
print(string1[1:4])
```

Output :

```
yth
```

เฉลยข้อที่ 27.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
string1 = 'Python'
print(string1[2:-1])
```

Output :

```
tho
```

เฉลยข้อที่ 28.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

4

Input :

```
string1 = 'Python'
print(string1[2:])
```

Output :

```
thon
```

เฉลยข้อที่ 29.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
string1 = 'Python'
print(string1[-4:5])
```

Output :

```
tho
```

เฉลยข้อที่ 30.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
string1 = 'Python'
print(string1[:5])
```

Output :

```
Pytho
```

วิธีที่ 2.

Input :

```
string1 = 'Python'
print(string1[-6:5])
```

Output :

```
Pytho
```

เฉลยข้อที่ 31.

P	y	t	h	o	n
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
string1 = 'Python'
print(string1[-4:-1])
```

Output :

```
tho
```

เฉลยข้อที่ 32.

Input :

```
string1 = input('Please insert string1: ')
print(len(string1))
```

Output :

```
Please insert string1: hello
5
```

เฉลยข้อที่ 33.

Input :

```
string1 = input('Please insert string1: ')
print('ก' in string1)
```

Output :

```
Please insert string1: การเรียน
True
```

4

เฉลยข้อที่ 34.

Input :

```
string1 = input('Please insert string1: ')
string2 = input('Please insert string2: ')
print(string2 in string1)
```

Output :

```
Please insert string1: abcde
Please insert string2: cd
True
```

เฉลยข้อที่ 35.

Input :

```
string1 = input('Please insert string1: ')
print(string1.replace('a', 'A'))
```

Output :

```
Please insert string1: abcd
Abcd
```

เฉลยข้อที่ 36.

Input :

```
string1 = input('Please insert string1: ')
string2 = string1.replace('a', 'A')
print(string2)
```

Output :

```
Please insert string1: aaa
AAA
```

เฉลยข้อที่ 37.

Input :

```
sentence = input('Please insert sentence: ')
print(sentence.split(' '))
```

Output :

```
Please insert sentence: learning python is good
for you
['learning', 'python', 'is', 'good', 'for', 'you']
```

เฉลยข้อที่ 38.

Input :

```
sentence = input('Please insert sentence: ')
c = input('Please insert c: ')
print(sentence.split(c))
```

Output :

```
Please insert sentence: one-two-three
Please insert c: -
['one', 'two', 'three']
```

เฉลยข้อที่ 39.

Input :

```
str1 = input('Please insert str1: ')
str2 = input('Please insert str2: ')
print(str1 + str2)
```

Output :

```
Please insert str1: ab
Please insert str2: cd
abcd
```

List

เฉลยข้อที่ 40.

0	1	2	'a'	'b'	'c'
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1)
print(type(list1))
```

Output :

```
[0, 1, 2, 'a', 'b', 'c']
<class 'list'>
```

เฉลยข้อที่ 41.

0	1	2	'a'	'b'	'c'
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[2])
```

Output :

```
2
```

เฉลยข้อที่ 42.

0	1	2	'a'	'b'	'c'
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[-3])
```

Output :

```
a
```

เฉลยข้อที่ 43.

0	1	2	'a'	'b'	'c'
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[0])
```

Output :

```
0
```

วิธีที่ 2.

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[-6])
```

Output :

```
0
```

เฉลยข้อที่ 44.

0	1	2	'a'	'b'	'c'
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[-2])
```

Output :

```
b
```

วิธีที่ 2.

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[4])
```

Output :

```
b
```

เฉลยข้อที่ 45.

0	1	2	'a'	'b'	'c'
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[2:4])
```

Output :

```
[2, 'a']
```

เฉลยข้อที่ 46.

0	1	2	'a'	'b'	'c'
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[1:-1])
```

Output :

```
[1, 2, 'a', 'b']
```

เฉลยข้อที่ 47.

0	1	2	'a'	'b'	'c'
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[1:5])
```

Output :

```
[1, 2, 'a', 'b']
```

วิธีที่ 2.

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[1:-1])
```

Output :

```
[1, 2, 'a', 'b']
```

วิธีที่ 3.

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[-5:5])
```

Output :

```
[1, 2, 'a', 'b']
```

วิธีที่ 4.

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[-5:-1])
```

Output :

```
[1, 2, 'a', 'b']
```

เฉลยข้อที่ 48.

0	1	2	'a'	'b'	'c'
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[3:])
```

Output :

```
['a', 'b', 'c']
```

วิธีที่ 2.

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[-3:])
```

Output :

```
['a', 'b', 'c']
```

เฉลยข้อที่ 49.

0	1	2	'a'	'b'	'c'
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[:3])
```

Output :

```
[0, 1, 2]
```


วิธีที่ 2.

Input :

```
list1 = [0, 1, 2, 'a', 'b', 'c']
print(list1[:-3])
```

Output :

```
[0, 1, 2]
```

เฉลยข้อที่ 50.

'ant'	'bird'	'cat'	'dog'	'eagle'
0	1	2	3	4
-5	-4	-3	-2	-1

➡

'ant'	'bird'	'cat'	'duck'	'eagle'
0	1	2	3	4
-5	-4	-3	-2	-1

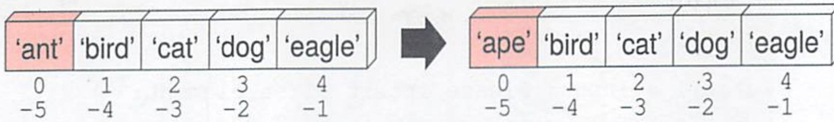
Input :

```
list2 = ['ant', 'bird', 'cat', 'dog', 'eagle']
list2[3] = 'duck'
print(list2)
```

Output :

```
print(list2)
['ant', 'bird', 'cat', 'duck', 'eagle']
```

เฉลยข้อที่ 51.



วิธีที่ 1. (recommend)

Input :

```
list2 = ['ant', 'bird', 'cat', 'dog', 'eagle']
list2[0] = 'ape'
print(list2)
```

Output :

```
['ape', 'bird', 'cat', 'dog', 'eagle']
```


วิธีที่ 2.

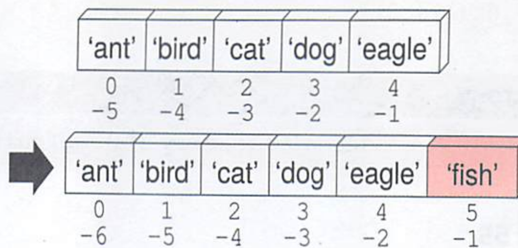
Input :

```
list2 = ['ant', 'bird', 'cat', 'dog', 'eagle']
list2[-5] = 'ape'
print(list2)
```

Output :

```
['ape', 'bird', 'cat', 'dog', 'eagle']
```

เฉลยข้อที่ 52.



Input :

```
list2 = ['ant', 'bird', 'cat', 'dog', 'eagle']
list2.append('fish')
print(list2)
```

Output :

```
['ant', 'bird', 'cat', 'dog', 'eagle', 'fish']
```

4

เฉลยข้อที่ 53.

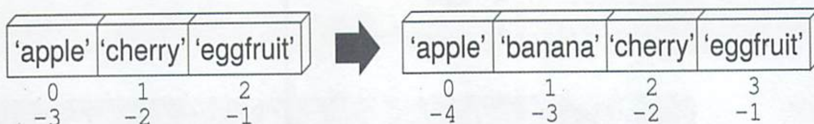
Input :

```
list2 = []
str1 = input('Please insert first element: ')
str2 = input('Please insert second element: ')
str3 = input('Please insert third element: ')
list2.append(str1)
list2.append(str2)
list2.append(str3)
print(list2)
```

Output :

```
Please insert first element: ant
Please insert second element: baboon
Please insert third element: camel
['ant', 'baboon', 'camel']
```

เฉลยข้อที่ 54.



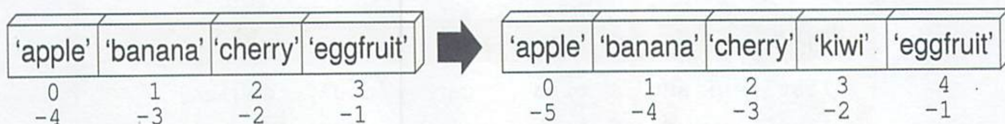
Input :

```
list3 = ['apple', 'cherry', 'eggfruit']
list3.insert(1, 'banana')
print(list3)
```

Output :

```
['apple', 'banana', 'cherry', 'eggfruit']
```

เฉลยข้อที่ 55.



Input :

```
list3 = ['apple', 'banana', 'cherry', 'eggfruit']
list3.insert(-2, 'kiwi')
print(list3)
```

Output :

```
['apple', 'banana', 'cherry', 'kiwi', 'eggfruit']
```

เฉลยข้อที่ 56.

Input :

```python
list4 = [0, 4, 2, 3, 1]
list4.sort()
print(list4)
```

Output :

```
[0, 1, 2, 3, 4]
```

เฉลยข้อที่ 57.

Input :

```python
list4 = [0, 4, 2, 3, 1]
list4.sort(reverse=True)
print(list4)
```

Output :

```
[4, 3, 2, 1, 0]
```

เฉลยข้อที่ 58.

Input :

```python
list4 = [0, 4, 2, 3, 1]
sorted_list4 = sorted(list4)
print(sorted_list4)
```

Output :

```
[0, 1, 2, 3, 4]
```

เฉลยข้อที่ 59.

Input :

```python
list4 = [0, 4, 2, 3, 1]
sorted_list4 = sorted(list4, reverse=True)
print(sorted_list4)
```

Output :

```
[4, 3, 2, 1, 0]
```

4

เฉลยข้อที่ 60.

Input :

```
list5 = ['d', 'a', 'c', 'b', 'e']
list5.sort()
print(list5)
```

Output :

```
['a', 'b', 'c', 'd', 'e']
```

เฉลยข้อที่ 61.

Input :

```
list5 = ['d', 'a', 'c', 'b', 'e']
sorted_list5 = sorted(list5, reverse=True)
print(sorted_list5)
```

Output :

```
['e', 'd', 'c', 'b', 'a']
```

เฉลยข้อที่ 62.

Input :

```
list1 = ['a', 'b', 'c']
list2 = [0, 1, 2]
list1.extend(list2)
print(list1)
print(list2)
```

Output :

```
['a', 'b', 'c', 0, 1, 2]
[0, 1, 2]
```

เฉลยข้อที่ 63.

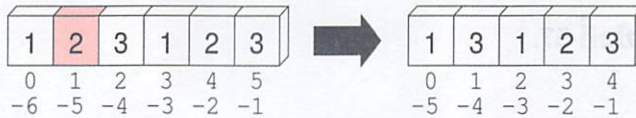
Input :

```
list1 = ['a', 'b', 'c']
list2 = [0, 1, 2]
list3 = list1 + list2
print(list1)
print(list2)
print(list3)
```

Output :

```
['a', 'b', 'c']
[0, 1, 2]
['a', 'b', 'c', 0, 1, 2]
```

เฉลยข้อที่ 64.



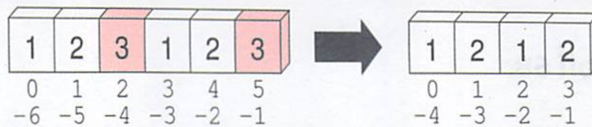
Input :

```
list1 = [1, 2, 3, 1, 2, 3]
del list1[1]
print(list1)
```

Output :

```
[1, 3, 1, 2, 3]
```

เฉลยข้อที่ 65.



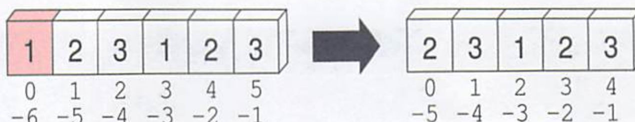
Input :

```
list1 = [1, 2, 3, 1, 2, 3]
del list1[-1]
del list1[2]
print(list1)
```

Output :

```
[1, 2, 1, 2]
```

เฉลยข้อที่ 66.



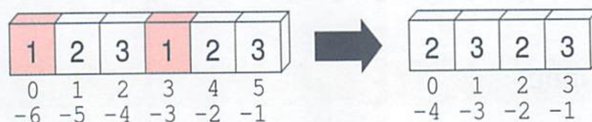
Input :

```python
list1 = [1, 2, 3, 1, 2, 3]
list1.remove(1)
print(list1)
```

Output :

```
[2, 3, 1, 2, 3]
```

เฉลยข้อที่ 67.



Input :

```python
list1 = [1, 2, 3, 1, 2, 3]
list1.remove(1)
list1.remove(1)
print(list1)
```

Output :

```
[2, 3, 2, 3]
```

เฉลยข้อที่ 68.

Input :

```python
list1 = [1, 2, 3, 1, 2, 3]
list1.clear()
print(list1)
```

Output :

```
[]
```

เฉลยข้อที่ 69.

Input :

```python
list1 = [1, 2, 3, 'a', 'b', 'c']
print(len(list1))
```

Output :

```
6
```

เฉลยข้อที่ 70.

Input :

```
list1 = [1, 2, 3, 4, 5]
numeric1 = int(input('Please insert numeric1: '))
print(numeric1 in list1)
```

Output :

```
Please insert numeric1: 4
True
```


Tuple

เฉลยข้อที่ 71.

'America'	'Brazil'	'China'	'Dominican'	'Egypt'
0	1	2	3	4
-5	-4	-3	-2	-1

Input :

```
tuple1 = ('America', 'Brazil', 'China', 'Dominican', 'Egypt')
print(tuple1)
print(type(tuple1))
```

Output :

```
('America', 'Brazil', 'China', 'Dominican',
'Egypt')
<class 'tuple'>
```

เฉลยข้อที่ 72.

'America'	'Brazil'	'China'	'Dominican'	'Egypt'
0	1	2	3	4
-5	-4	-3	-2	-1

Input :

```
tuple1 = ('America', 'Brazil', 'China', 'Dominican', 'Egypt')
print(tuple1[1])
```

Output :

```
Brazil
```

เฉลยข้อที่ 73.

'America'	'Brazil'	'China'	'Dominican'	'Egypt'
0	1	2	3	4
-5	-4	-3	-2	-1

Input :

```
tuple1 = ('America', 'Brazil', 'China',
    ↪ 'Dominican', 'Egypt')
print(tuple1[-2])
```

Output :

```
Dominican
```

เฉลยข้อที่ 74.

'America'	'Brazil'	'China'	'Dominican'	'Egypt'
0	1	2	3	4
-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
tuple1 = ('America', 'Brazil', 'China',
    ↪ 'Dominican', 'Egypt')
print(tuple1[2])
```

Output :

```
China
```

วิธีที่ 2.

Input :

```
tuple1 = ('America', 'Brazil', 'China',
    ↪ 'Dominican', 'Egypt')
print(tuple1[-3])
```

Output :

```
China
```

เฉลยข้อที่ 75.

'America'	'Brazil'	'China'	'Dominican'	'Egypt'
0	1	2	3	4
-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
tuple1 = ('America', 'Brazil', 'China',
    'Dominican', 'Egypt')
print(tuple1[-1])
```

Output :

```
Egypt
```

วิธีที่ 2.

Input :

```
tuple1 = ('America', 'Brazil', 'China',
    'Dominican', 'Egypt')
print(tuple1[4])
```

Output :

```
Egypt
```

เฉลยข้อที่ 76.

'one'	'two'	'three'	1	2	3
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
tuple2 = ('one', 'two', 'three', 1, 2, 3)
print(tuple2[1:4])
```

Output :

```
('two', 'three', 1)
```

เฉลยข้อที่ 77.

'one'	'two'	'three'	1	2	3
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

Input :

```
tuple2 = ('one', 'two', 'three', 1, 2, 3)
print(tuple2[2:-1])
```

Output :

```
('three', 1, 2)
```

เฉลยข้อที่ 78.

'one'	'two'	'three'	1	2	3
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
tuple2 = ('one', 'two', 'three', 1, 2, 3)
print(tuple2[2:4])
```

Output :

```
('three', 1)
```

วิธีที่ 2.

Input :

```
tuple2 = ('one', 'two', 'three', 1, 2, 3)
print(tuple2[2:-2])
```

Output :

```
('three', 1)
```

วิธีที่ 3.

Input :

```
tuple2 = ('one', 'two', 'three', 1, 2, 3)
print(tuple2[-4:4])
```

Output :

```
('three', 1)
```

วิธีที่ 4.

Input :

```
tuple2 = ('one', 'two', 'three', 1, 2, 3)
print(tuple2[-4:-2])
```

Output :

```
('three', 1)
```

4

เฉลยข้อที่ 79.

'one'	'two'	'three'	1	2	3
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
tuple2 = ('one', 'two', 'three', 1, 2, 3)
print(tuple2[:3])
```

Output :

```
('one', 'two', 'three')
```

วิธีที่ 2.

Input :

```
tuple2 = ('one', 'two', 'three', 1, 2, 3)
print(tuple2[:-3])
```

Output :

```
('one', 'two', 'three')
```

เฉลยข้อที่ 80.

'one'	'two'	'three'	1	2	3
0	1	2	3	4	5
-6	-5	-4	-3	-2	-1

วิธีที่ 1. (recommend)

Input :

```
tuple2 = ('one', 'two', 'three', 1, 2, 3)
print(tuple2[3:])
```

Output :

```
(1, 2, 3)
```

4

วิธีที่ 2.

Input :

```
tuple2 = ('one', 'two', 'three', 1, 2, 3)
print(tuple2[-3:])
```

Output :

```
(1, 2, 3)
```

เฉลยข้อที่ 81.

Input :

```
tuple3 = ('one', 'two', 'three', 'four', 'five')
print(len(tuple3))
```

Output :

```
5
```

เฉลยข้อที่ 82.

Input :

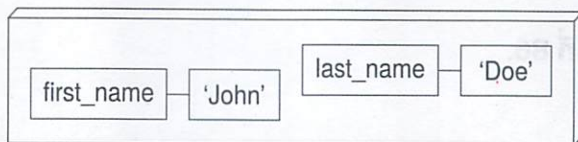
```
tuple3 = ('one', 'two', 'three', 'four', 'five')
str1 = input('Please insert str1: ')
print(str1 in tuple3)
```

Output :

```
Please insert str1: three
True
```

Dictionary

เฉลยข้อที่ 83.



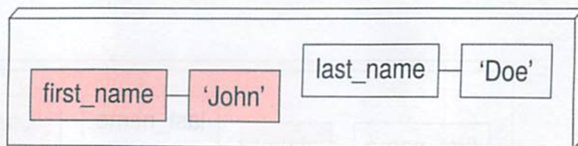
Input :

```python
dict1 = {'first_name' : 'John', 'last_name' : 'Doe'}
print(dict1)
print(type(dict1))
```

Output :

```
{'first_name': 'John', 'last_name': 'Doe'}
<class 'dict'>
```

เฉลยข้อที่ 84.



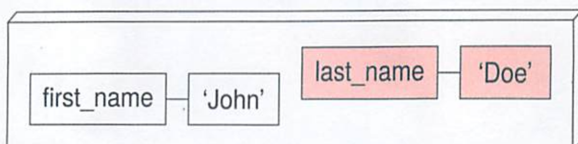
Input :

```python
dict1 = {'first_name' : 'John', 'last_name' : 'Doe'}
print(dict1['first_name'])
```

Output :

```
John
```

เฉลยข้อที่ 85.



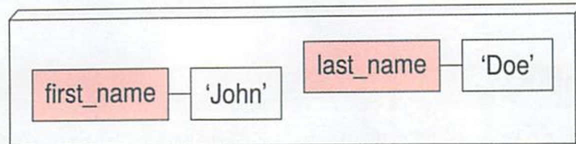
Input :

```
dict1 = {'first_name' : 'John', 'last_name' : 'Doe'}
print(dict1['last_name'])
```

Output :

```
Doe
```

เฉลยข้อที่ 86.



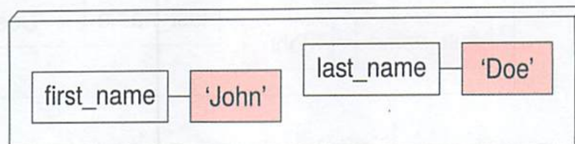
Input :

```
dict1 = {'first_name' : 'John', 'last_name' : 'Doe'}
print(dict1.keys())
```

Output :

```
dict_keys(['first_name', 'last_name'])
```

เฉลยข้อที่ 87.



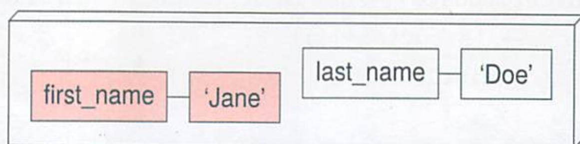
Input :

```
dict1 = {'first_name' : 'John', 'last_name' : 'Doe'}
print(dict1.values())
```

Output :

```
dict_values(['John', 'Doe'])
```

เฉลยข้อที่ 88.



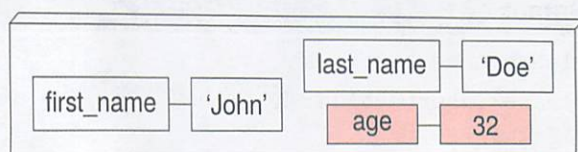
Input :

```
dict1 = {'first_name' : 'John', 'last_name' : 'Doe'}
dict1['first_name'] = 'Jane'
print(dict1)
```

Output :

```
{'first_name': 'Jane', 'last_name': 'Doe'}
```

เฉลยข้อที่ 89.



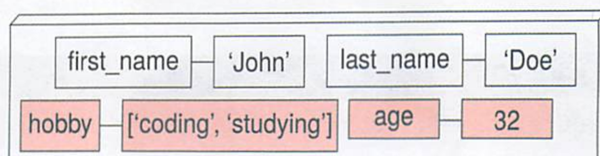
Input :

```
dict1 = {'first_name' : 'John', 'last_name' : 'Doe'}
dict1['age'] = 32
print(dict1)
```

Output :

```
{'first_name': 'John', 'last_name': 'Doe', 'age':
32}
```

เฉลยข้อที่ 90.



Input :

```
dict1 = {'first_name' : 'John', 'last_name' : 'Doe'}
dict1.update({'age' : 32, 'hobby' : ['coding'
    ↪ 'studying']})
print(dict1)
```

Output :

```
{'first_name': 'John', 'last_name': 'Doe', 'age':
32, 'hobby': ['coding', 'studying']}
```

4

เฉลยข้อที่ 91.

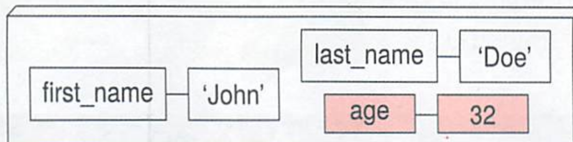
Input :

```
dict1 = {}
str1 = input('Please insert str1: ')
str2 = input('Please insert str2: ')
dict1[str1] = str2
print(dict1)
```

Output :

```
Please insert key: a
Please insert value: A
{'a': 'A'}
```

เฉลยข้อที่ 92.



Input :

```
dict1 = {'first_name' : 'John', 'last_name' : 'Doe',
    ↪ 'age' : 32}
del dict1['age']
print(dict1)
```

Output :

```
{'first_name': 'John', 'last_name': 'Doe'}
```

เฉลยข้อที่ 93.

Input :

```
dict1 = {'first_name' : 'John', 'last_name' : 'Doe',
    ↪ 'age' : 32}
dict1.clear()
print(dict1)
```

Output :

```
{}
```

4

เฉลยข้อที่ 94.

วิธีที่ 1. (recommend)

Input :

```
dict1 = {'first_name' : 'John', 'last_name' :
    ↪ 'Doe', 'age' : 32}
print(len(dict1))
```

Output :

```
3
```

วิธีที่ 2.

Input :

```
dict1 = {'first_name' : 'John', 'last_name' :
    ↪ 'Doe', 'age' : 32}
print(len(dict1.keys()))
```

Output :

```
3
```

เฉลยข้อที่ 95.

วิธีที่ 1. (recommend)

Input :

```
dict1 = {'first_name' : 'John', 'last_name' :
    ↪ 'Doe', 'age' : 32}
str1 = input('Please insert str1: ')
print(str1 in dict1)
```

Output :

```
Please insert str1: first_name
True
```

วิธีที่ 2.

Input :

```
dict1 = {'first_name' : 'John', 'last_name' :
    ↪ 'Doe', 'age' : 32}
str1 = input('Please insert str1: ')
print(str1 in dict1.keys())
```

Output :

```
Please insert str1: first_name
True
```

4

เฉลยข้อที่ 96.

Input :

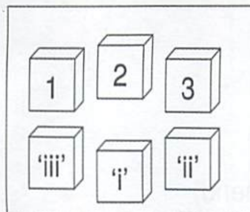
```
dict1 = {'first_name' : 'John', 'last_name' : 'Doe',
    ↪ 'age' : 32}
str1 = input('Please insert str1: ')
print(str1 in dict1.values())
```

Output :

```
Please insert str1: Doe
True
```

Set

เฉลยข้อที่ 97.



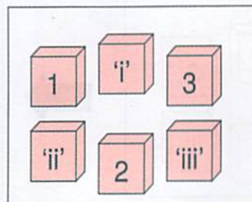
Input :

```
set1 = {1, 2, 3, 'i', 'ii', 'iii'}
print(set1)
print(type(set1))
```

Output :

```
{1, 2, 3, 'iii', 'i', 'ii'}
<class 'set'>
```

เฉลยข้อที่ 98.



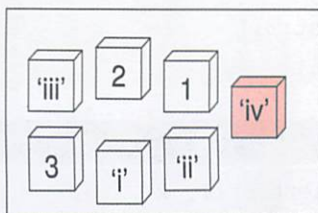
Input :

```
set1 = {1, 2, 3, 'i', 'ii', 'iii'}
for s in set1:
    print(s)
```

Output :

```
1
2
3
iii
i
ii
```

เฉลยข้อที่ 99.



Input :

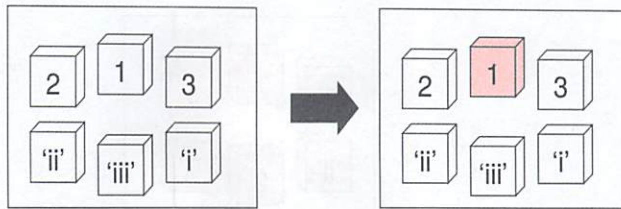
```
set1 = {1, 2, 3, 'i', 'ii', 'iii'}
set1.add('iv')
print(set1)
```

Output :

```
{1, 2, 3, 'iii', 'iv', 'i', 'ii'}
```

4

เฉลยข้อที่ 100.



Input :

```
set1 = {1, 2, 3, 'i', 'ii', 'iii'}
set1.add(1)
print(set1)
```

Output :

```
{1, 2, 3, 'iii', 'i', 'ii'}
```

เฉลยข้อที่ 101.

Input :

```
set1 = set()
str1 = input('Please insert str1: ')
str2 = input('Please insert str2: ')
str3 = input('Please insert str3: ')
set1.add(str1)
set1.add(str2)
set1.add(str3)
print(set1)
```

Output :

```
Please insert str1: x
Please insert str2: y
Please insert str3: z
{'x', 'y', 'z'}
```

เฉลยข้อที่ 102.

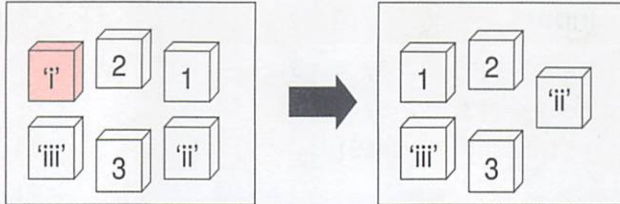
Input :

```
set1 = set()
str1 = input('Please insert str1: ')
str2 = input('Please insert str2: ')
str3 = input('Please insert str3: ')
set1.update({str1, str2, str3})
print(set1)
```

Output :

```
Please insert str1: 1
Please insert str2: a
Please insert str3: c
{'c', 'a', '1'}
```

เฉลยข้อที่ 103.



Input :

```
set1 = {1, 2, 3, 'i', 'ii', 'iii'}
set1.remove('i')
print(set1)
```

Output :

```
{1, 2, 3, 'iii', 'ii'}
```

เฉลยข้อที่ 104.

Input :

```
set1 = {1, 2, 3, 'i', 'ii', 'iii'}
print(len(set1))
```

Output :

```
6
```

เฉลยข้อที่ 105.

Input :

```
set1 = {1, 2, 3, 'i', 'ii', 'iii'}
int1 = int(input('Please insert int1: '))
print(int1 in set1)
```

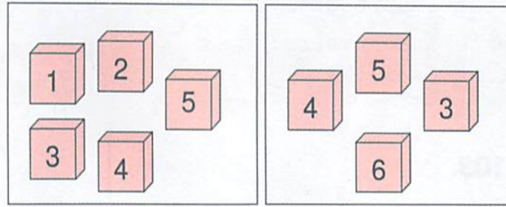
Output :

```
Please insert numeric1: 2
True
```

4

4

เฉลยข้อที่ 106.



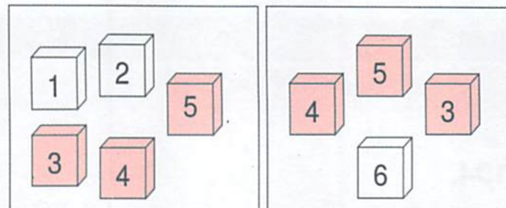
Input :

```
set1 = {1, 2, 3, 4, 5}
set2 = {3, 4, 5, 6}
print(set1 | set2)
```

Output :

```
{1, 2, 3, 4, 5, 6}
```

เฉลยข้อที่ 107.



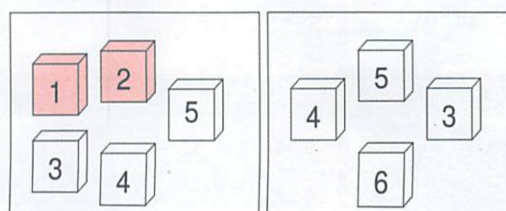
Input :

```
set1 = {1, 2, 3, 4, 5}
set2 = {3, 4, 5, 6}
print(set1 & set2)
```

Output :

```
{3, 4, 5}
```

เฉลยข้อที่ 108.



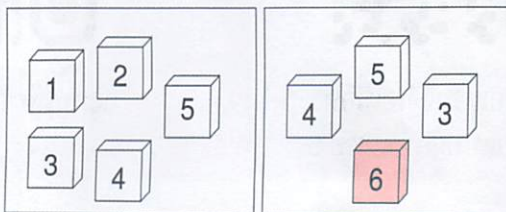
Input :

```
set1 = {1, 2, 3, 4, 5}
set2 = {3, 4, 5, 6}
print(set1 - set2)
```

Output :

```
{1, 2}
```

เฉลยข้อที่ 109.



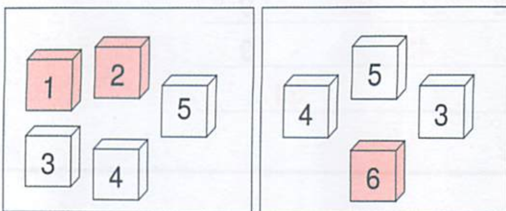
Input :

```
set1 = {1, 2, 3, 4, 5}
set2 = {3, 4, 5, 6}
print(set2 - set1)
```

Output :

```
{6}
```

เฉลยข้อที่ 110.



Input :

```
set1 = {1, 2, 3, 4, 5}
set2 = {3, 4, 5, 6}
print(set1 ^ set2)
```

Output :

```
{1, 2, 6}
```

05 OPERATOR


สแกนเพื่อเรียนวิดีโอเนื้อหา
และเฉลยแบบฝึกหัดบทที่ 5


สแกนเพื่อรับ code บทที่ 5

5.1 Problem

จงเขียนลำดับการคำนวณของนิพจน์ต่อไปนี้

Ex. 2**3+5*7-9

2**3	+	5*7	-	9
8	+	5*7	-	9
8	+	35	-	9
	43		-	9
			34	

1. 2**3 + 5*7 - 9

2. 5*2**3 + 4*5

3. 2**3**2

4. 5 + 4 + 3 + 2 + 1

5. 7%2**2

6. 7//(3**2**2 - 75)

7. 6%7//3 + 2**3

8. 5**2//(2*2 + 4)

5

9. 1 + 2**3*1 - 1/2

10. (1 + 2) + (3*4) - (2**3)

11. y = 2**2

12. y = 3**2/2**3

13. y = 2**4**1/2

14. y = 2*(4**2)%(8//2)

15. y = 2*4 - 15 >= 1

5.2 Solution

เฉลยข้อที่ 1.

```
2**3 + 5*7 - 9
 8   + 5*7 - 9
 8   +  35 - 9
     43    - 9
           34
```

Input :

```
2**3 + 5*7 - 9
```

Output :

```
34
```

เฉลยข้อที่ 2.

```
5*2**3 + 4*5
5* 8   + 4*5
 40    + 4*5
 40    +  20
       60
```

Input :

```
5*2**3 + 4*5
```

Output :

```
60
```

เฉลยข้อที่ 3.

```
2**3**2
2** 9
 512
```

Input :

```
2**3**2
```

Output :

```
512
```

เฉลยข้อที่ 4.

```
5 + 4 + 3 + 2 + 1
  9   + 3 + 2 + 1
       12   + 2 + 1
             14 + 1
                15
```

Input :

```
5 + 4 + 3 + 2 + 1
```

Output :

```
15
```

เฉลยข้อที่ 5.

```
7%2**2
7% 4
 3
```

Input :

```
7%2**2
```

Output :

```
3
```

เฉลยข้อที่ 6.

```
7//(3**2**2 - 75)
7//(3** 4   - 75)
7//( 81     - 75)
7//(          6)
 1
```

Input :

```
7//(3**2**2 - 75)
```

Output :

```
1
```

5

เฉลยข้อที่ 7.

```
6%7//3 + 2**3
6%7//3 +  8
 6 //3 +  8
   2   +  8
      10
```

Input :

```
6%7//3 + 2**3
```

Output :

```
10
```

เฉลยข้อที่ 8.

```
5**2//(2*2 + 4)
 25 //(2*2 + 4)
 25 //( 4  + 4)
 25 //      8
    3
```

Input :

```
5**2//(2*2 + 4)
```

Output :

```
3
```

เฉลยข้อที่ 9.

```
1 + 2**3*1 - 1/2
1 +  8  *1 - 1/2
1 +      8 - 1/2
1 +      8 - 0.5
  9          - 0.5
             8.5
```

Input :

```
1 + 2**3*1 - 1/2
```

Output :

```
8.5
```

เฉลยข้อที่ 10.

```
(1 + 2) + (3*4) - (2**3)
    3   + (3*4) - (2**3)
    3   +   12  - (2**3)
    3   +   12  -    8
       15       -    8
                7
```

Input :

```
(1 + 2) + (3*4) -
    ↪ (2**3)
```

Output :

```
7
```

เฉลยข้อที่ 11.

```
y = 2**2
y =  4
```

Input :

```
y = 2**2
print(y)
```

Output :

```
4
```

เฉลยข้อที่ 12.

```
y = 3**2/2**3
y =  9  /2**3
y =  9  / 8
y =    1.125
```

Input :

```
y = 3**2/2**3
print(y)
```

Output :

```
1.125
```

เฉลยข้อที่ 13.

```
y = 2**4**1/2
y = 2** 4  /2
y =  16    /2
y =          8
```

Input :

```
y = 2**4**1/2
print(y)
```

Output :

```
8.0
```

เฉลยข้อที่ 14.

```
y = 2*(4**2)%(8//2)
y = 2*  16  %(8//2)
y = 2*  16  %  4
y =  32     %  4
y =            0
```

Input :

```
y = 2*(4**2)%(8//2)
print(y)
```

Output :

```
0
```

เฉลยข้อที่ 15.

```
y = 2*4 - 15 >= 1
y =  8  - 15 >= 1
y =      -7  >= 1
y =             False
```

Input :

```
y = 2*4 - 15 >= 1
print(y)
```

Output :

```
False
```

06 IF-ELSE (FLOWCHART-CODE)



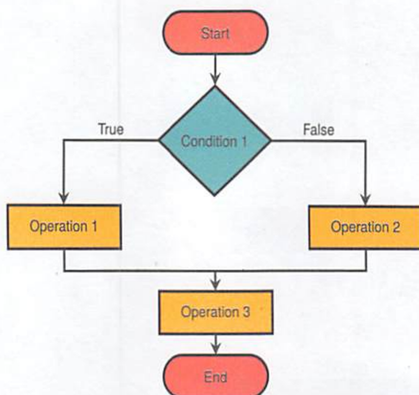
สแกนเพื่อเรียนวิดีโอเนื้อหาและเฉลยแบบฝึกหัดบทที่ 6

6.1 Problem

เขียน flowchart จาก code ที่กำหนด

Ex.

```python
if condition1:
    operation1
else:
    operation2
operation3
```



1.

```
if condition1:
    operation1
else:
    operation2
operation3
```

2.

```
if condition1:
    operation1
else:
    operation2
```

3.

```
if condition1:
    operation1
operation2
```

4.

```
if condition1:
    operation1
```

5.

```
if condition1:
    operation1
elif condition2:
    operation2
else:
    operation3
operation4
```

6.

```
if condition1:
    operation1
elif condition2:
    operation2
else:
    operation3
```

7.

```
if condition1:
    operation1
    operation2
elif condition2:
    operation3
elif condition3:
    operation4
else:
    operation5
operation6
```

8.

```
if condition1:
    operation1
elif condition2:
    operation2
elif condition3:
    operation3
else:
    operation4
```

9.

```
if condition1:
    operation1
elif condition2:
    operation2
operation3
```

10.

```
if condition1:
    operation1
    operation2
elif condition2:
    operation3
```

6

11.

```
if condition1:
    operation1
elif condition2:
    operation2
elif condition3:
    operation3
operation4
```

12.

```
if condition1:
    operation1
    operation2
elif condition2:
    operation3
elif condition3:
    operation4
```

13.

```
if condition1:
    if condition2:
        operation1
    operation2
else:
    operation3
    if condition3:
        operation4
    elif condition4:
        operation5
        operation6
operation7
```

14.

```
if condition1:
    operation1
    if condition2:
        operation2
    else:
        operation3
    operation4
    operation5
else:
    if condition3:
        operation6
    else:
        operation7
operation8
```

15.

```
if condition1:
    if condition2:
        operation1
    elif condition3:
        operation2
    else:
        operation3
else:
    operation4
    if condition4:
        operation5
        operation6
    operation7
    operation8
operation9
```

16.

```
if condition1:
    operation1
    if condition2:
        operation2
    elif condition3:
        operation3
    else:
        operation4
else:
    if condition4:
        operation5
    elif condition5:
        operation6
    operation7
operation8
```

17.

```
if condition1:
    operation1
    operation2
    if condition2:
        operation3
    operation4
operation5
```

18.

```
if condition1:
    operation1
    if condition2:
        operation2
    else:
        operation3
operation4
operation5
```

19.

```
if condition1:
    if condition2:
        operation1
elif condition3:
    operation2
    if condition4:
        operation3
    elif condition5:
        operation4
    operation5
else:
    if condition6:
        operation6
operation7
operation8
```

20.

```
if condition1:
    if condition2:
        operation1
    else:
        operation2
elif condition3:
    operation3
    if condition4:
        operation4
        operation5
    elif condition5:
        operation6
    operation7
    operation8
else:
    if condition6:
        operation9
        operation10
    elif condition7:
        operation11
    operation12
operation13
```

6

6

21.

```
if condition1:
    if condition2:
        operation1
        operation2
    elif condition3:
        operation3
    else:
        operation4
elif condition4:
    if condition5:
        operation5
        operation6
    elif condition6:
        operation7
else:
    operation8
    operation9
    if condition7:
        operation10
        operation11
    else:
        operation12
        operation13
operation14
```

22.

```
if condition1:
    operation1
    if condition2:
        operation2
    operation3
elif condition3:
    operation4
    if condition4:
        operation5
    elif condition5:
        operation6
    else:
        operation7
        operation8
    operation9
else:
    if condition6:
        operation10
    elif condition7:
        operation11
operation12
```

23.

```
if condition1:
    if condition2:
        operation1
    elif condition3:
        operation2
    else:
        operation3
elif condition4:
    if condition5:
        operation4
    else:
        operation5
        operation6
else:
    if condition6:
        operation7
operation8
```

24.

```
if condition1:
    if condition2:
        operation1
elif condition3:
    operation2
    if condition4:
        operation3
    elif condition5:
        operation4
    else:
        operation5
    operation6
else:
    if condition6:
        operation7
        operation8
operation9
operation10
```

25.

```
if condition1:
    operation1
    operation2
    if condition2:
        operation3
    operation4
elif condition3:
    operation5
    operation6
    if condition4:
        operation7
    operation8
operation9
```

26.

```
if condition1:
    operation1
    operation2
    if condition2:
        operation3
        operation4
    elif condition3:
        operation5
    else:
        operation6
    operation7
    operation8
elif condition4:
    operation9
    if condition5:
        operation10
    elif condition6:
        operation11
        operation12
    operation13
operation14
```

27.

```
if condition1:
    if condition2:
        operation1
    else:
        operation2
elif condition3:
    if condition4:
        operation3
    elif condition5:
        operation4
    operation5
operation6
```

6

6

28.

```
if condition1:
    operation1
    if condition2:
        operation2
    operation3
elif condition3:
    operation4
    if condition4:
        operation5
operation6
```

29.

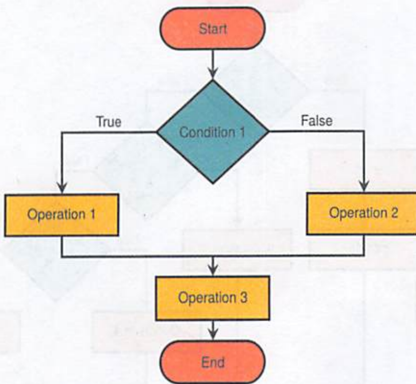
```
if condition1:
    if condition2:
        if condition3:
            operation1
            operation2
        else:
            operation3
        operation4
    else:
        operation5
        if condition4:
            operation6
            operation7
    operation8
else:
    operation9
    if condition5:
        if condition6:
            operation10
    else:
        if condition7:
            operation11
        operation12
    operation13
operation14
operation15
```

30.

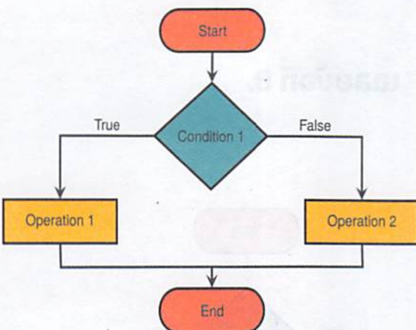
```
if condition1:
    operation1
    if condition2:
        if condition3:
            operation2
            operation3
        operation4
    else:
        if condition4:
            operation5
        else:
            operation6
    operation7
operation8
```

6.2 Solution

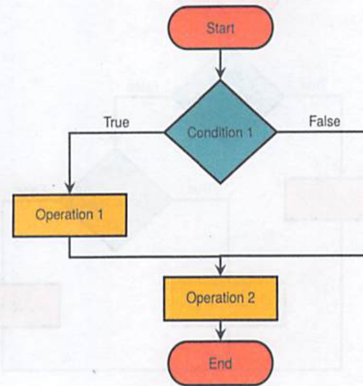
เฉลยข้อที่ 1.



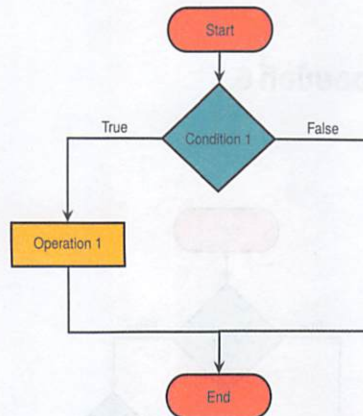
เฉลยข้อที่ 2.



เฉลยข้อที่ 3.

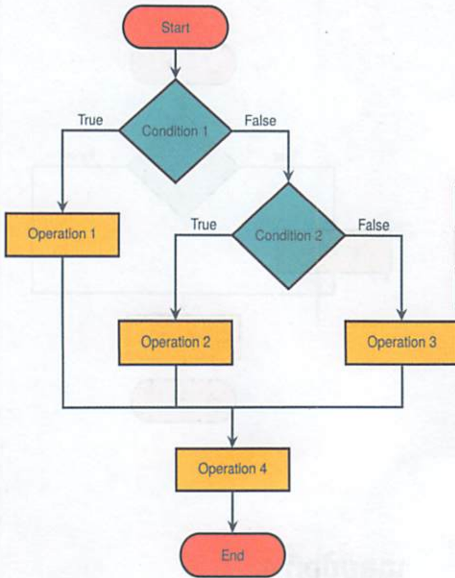


เฉลยข้อที่ 4.

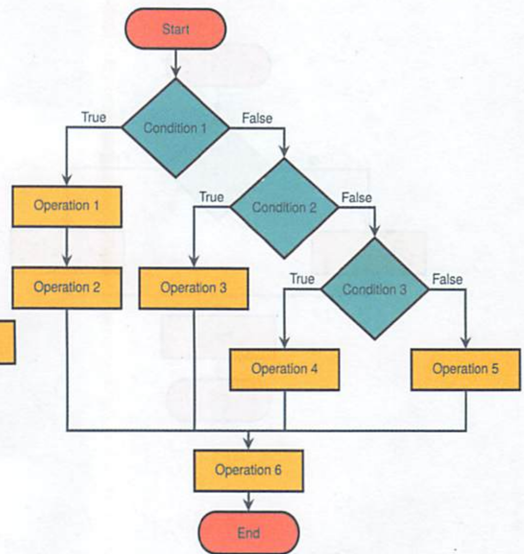


6

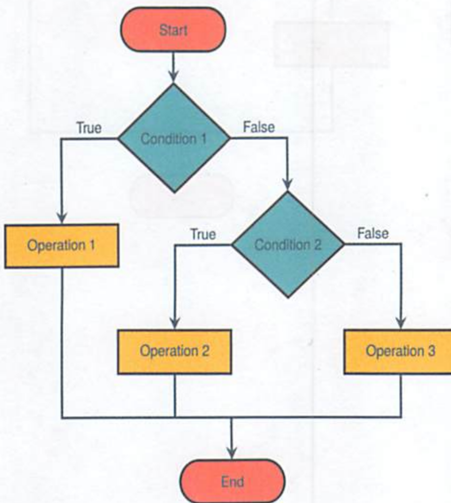
เฉลยข้อที่ 5.



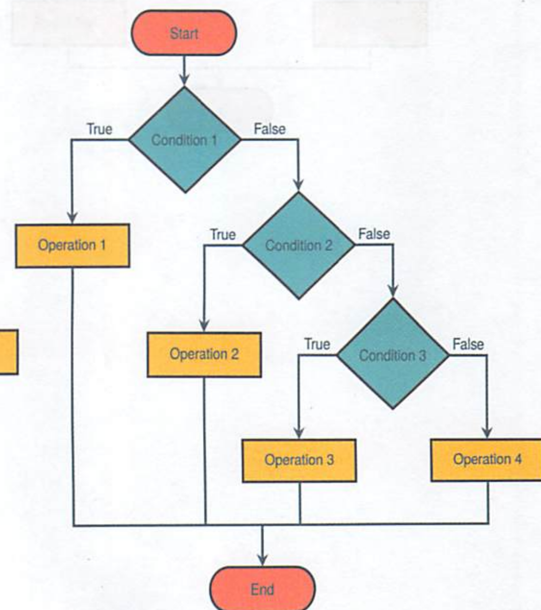
เฉลยข้อที่ 7.



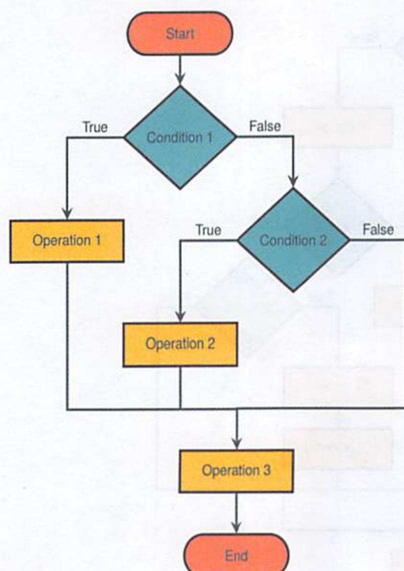
เฉลยข้อที่ 6.



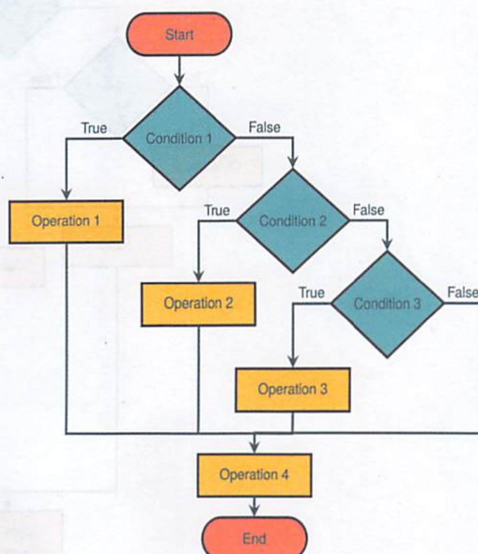
เฉลยข้อที่ 8.



เฉลยข้อที่ 9.

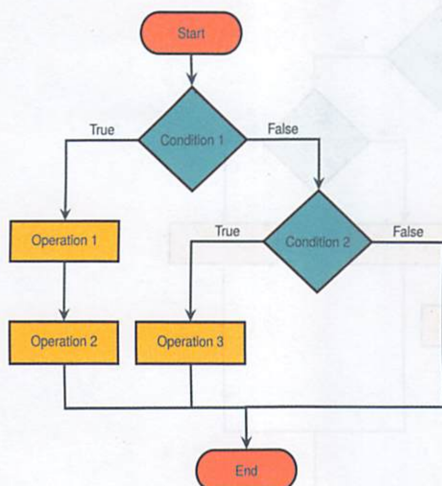


เฉลยข้อที่ 11.

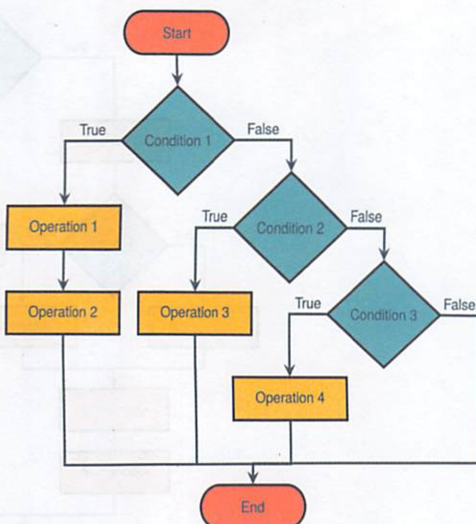


6

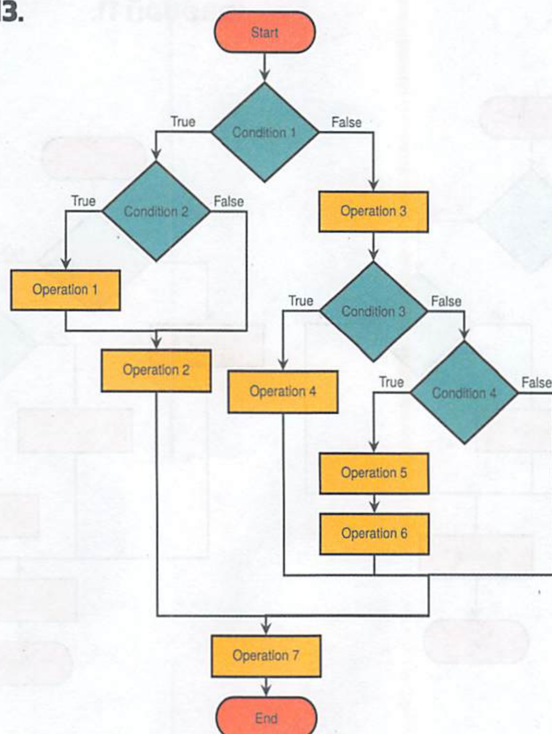
เฉลยข้อที่ 10.



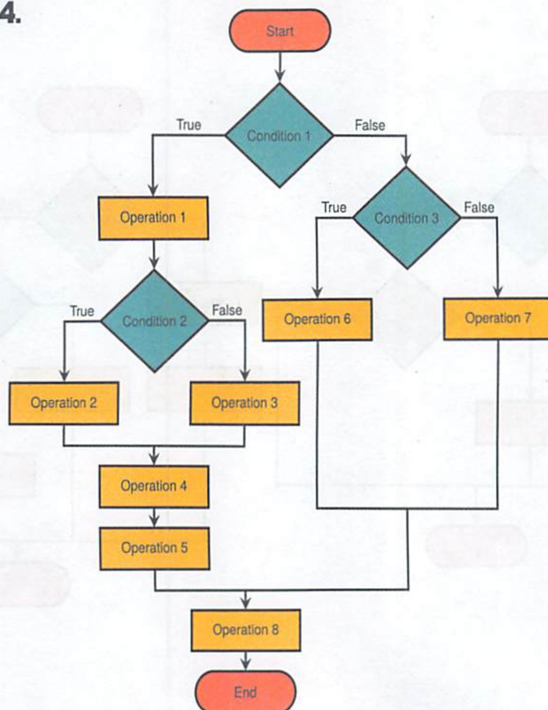
เฉลยข้อที่ 12.



เฉลยข้อที่ 13.

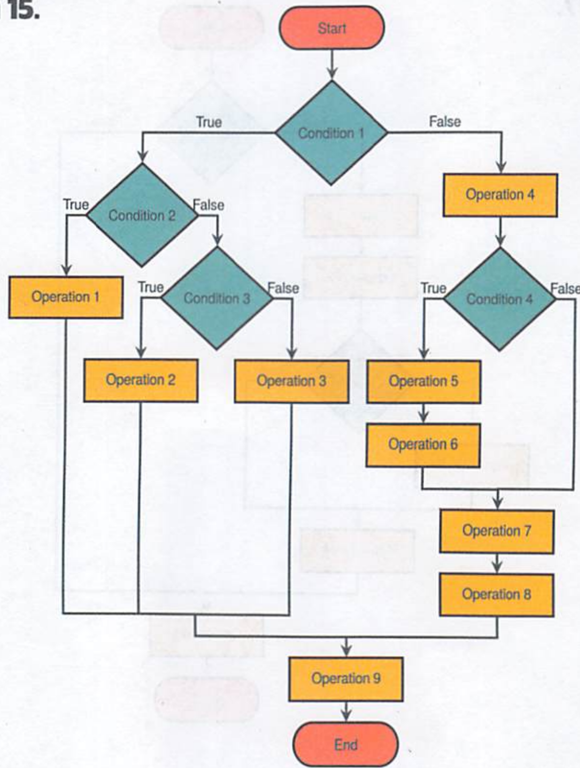


เฉลยข้อที่ 14.



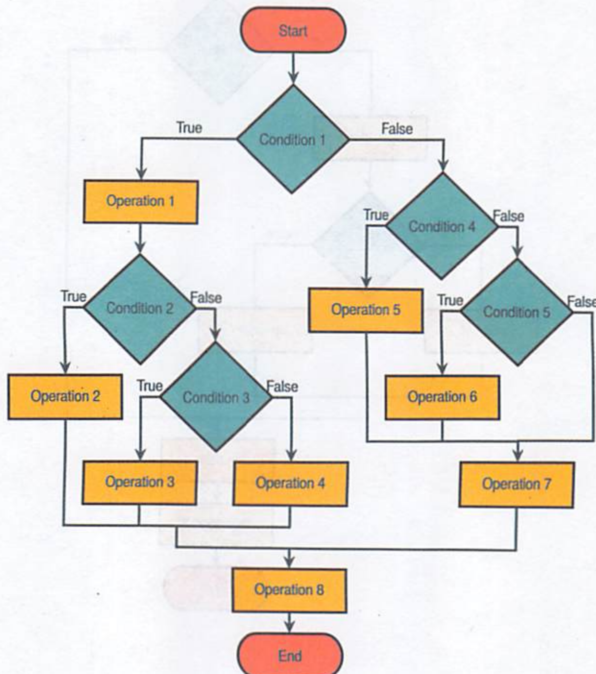
6

เฉลยข้อที่ 15.



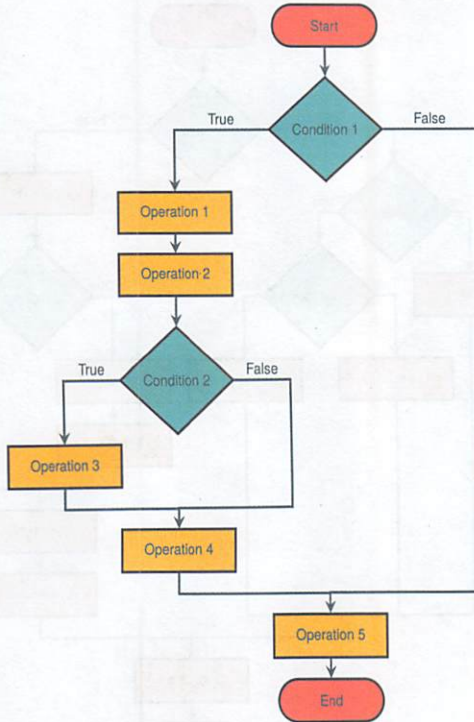
6

เฉลยข้อที่ 16.

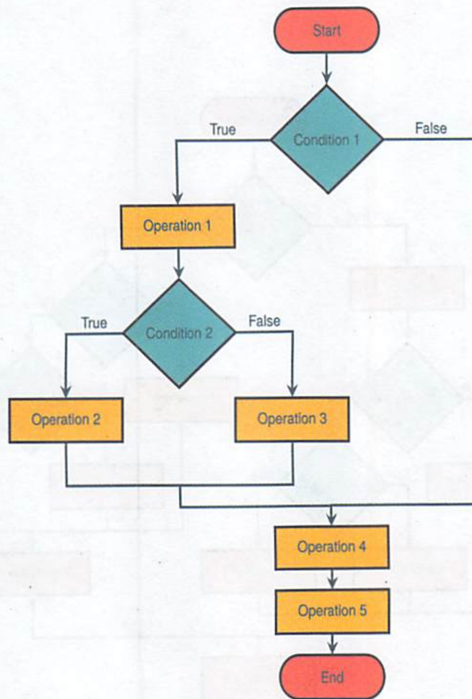


6

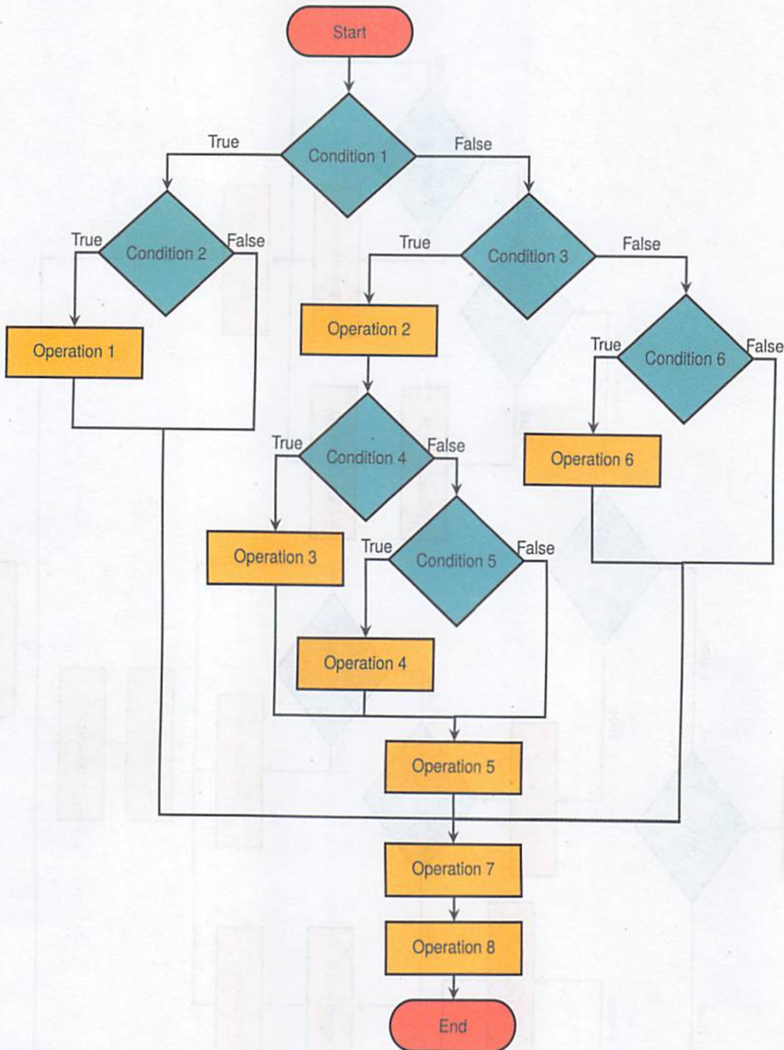
เฉลยข้อที่ 17.



เฉลยข้อที่ 18.

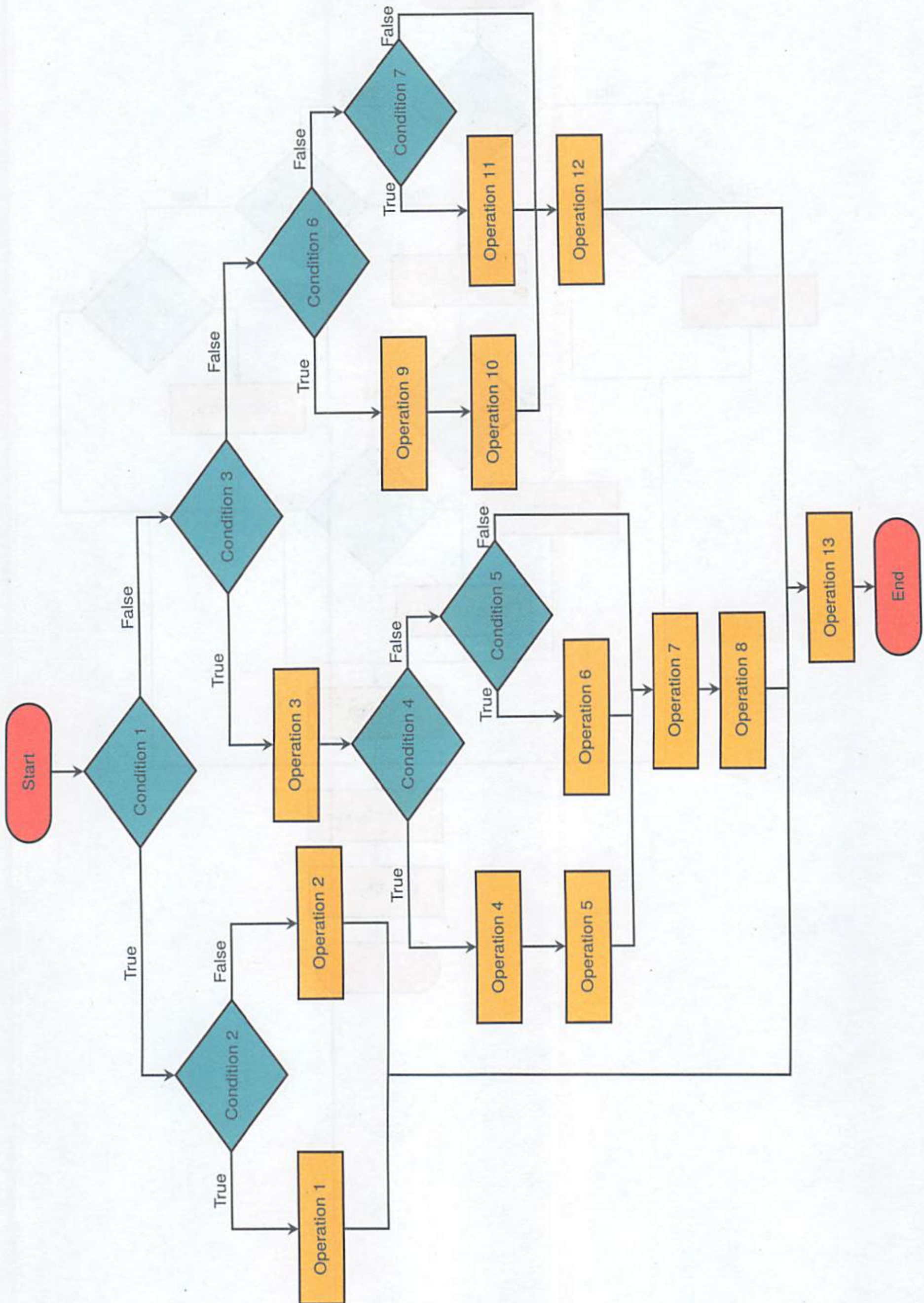


เฉลยข้อที่ 19.

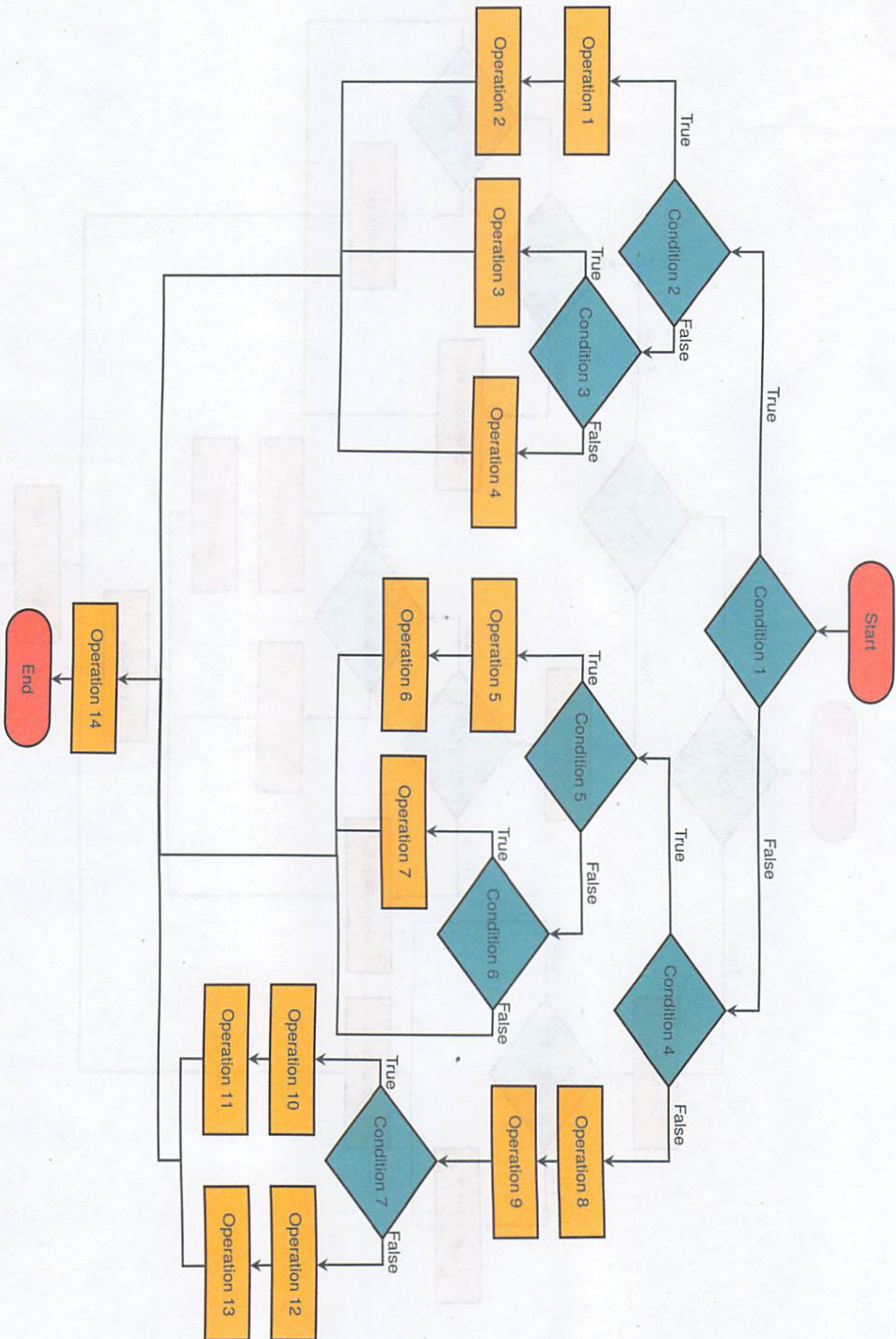

Start
Condition 1
True
False
Condition 2
True
False
Operation 1
Condition 3
True
False
Operation 2
Condition 6
True
False
Operation 6
Condition 4
True
False
Operation 3
Condition 5
True
False
Operation 4
Operation 5
Operation 7
Operation 8
End

6

เฉลยข้อที่ 20.


Start
Condition 1
True
False
Condition 2
True
False
Operation 1
Operation 2
Condition 3
True
False
Operation 3
Condition 4
True
False
Operation 4
Operation 5
Condition 5
True
False
Operation 6
Operation 7
Operation 8
Condition 6
True
False
Operation 9
Operation 10
Condition 7
True
False
Operation 11
Operation 12
Operation 13
End

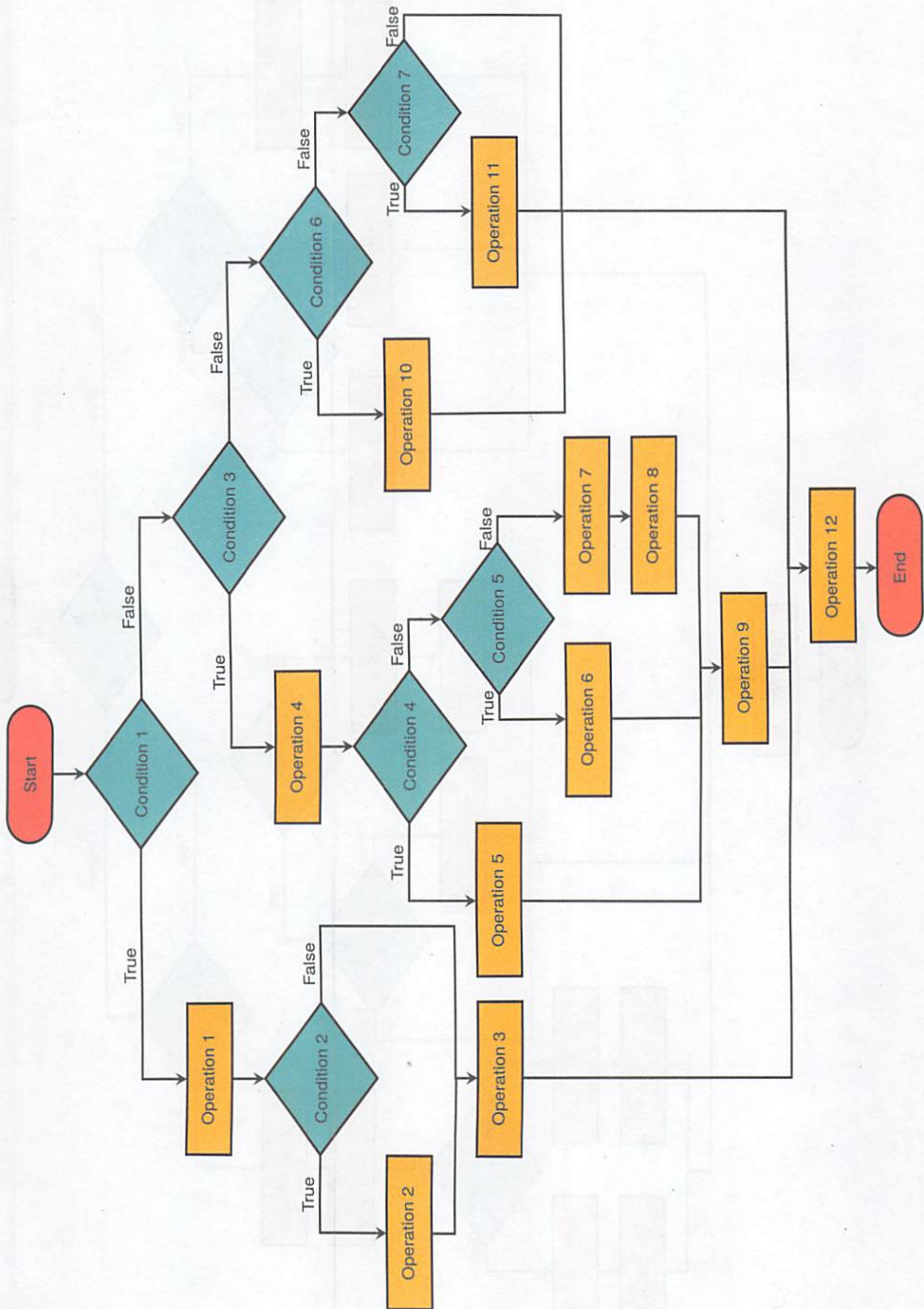
เฉลยข้อที่ 21.


Start
Condition 1
True
False
Condition 2
True
False
Operation 1
Operation 2
Condition 3
True
False
Operation 3
Operation 4
Condition 4
True
False
Condition 5
True
False
Operation 5
Operation 6
Condition 6
True
False
Operation 7
Operation 8
Operation 9
Condition 7
True
False
Operation 10
Operation 11
Operation 12
Operation 13
Operation 14
End

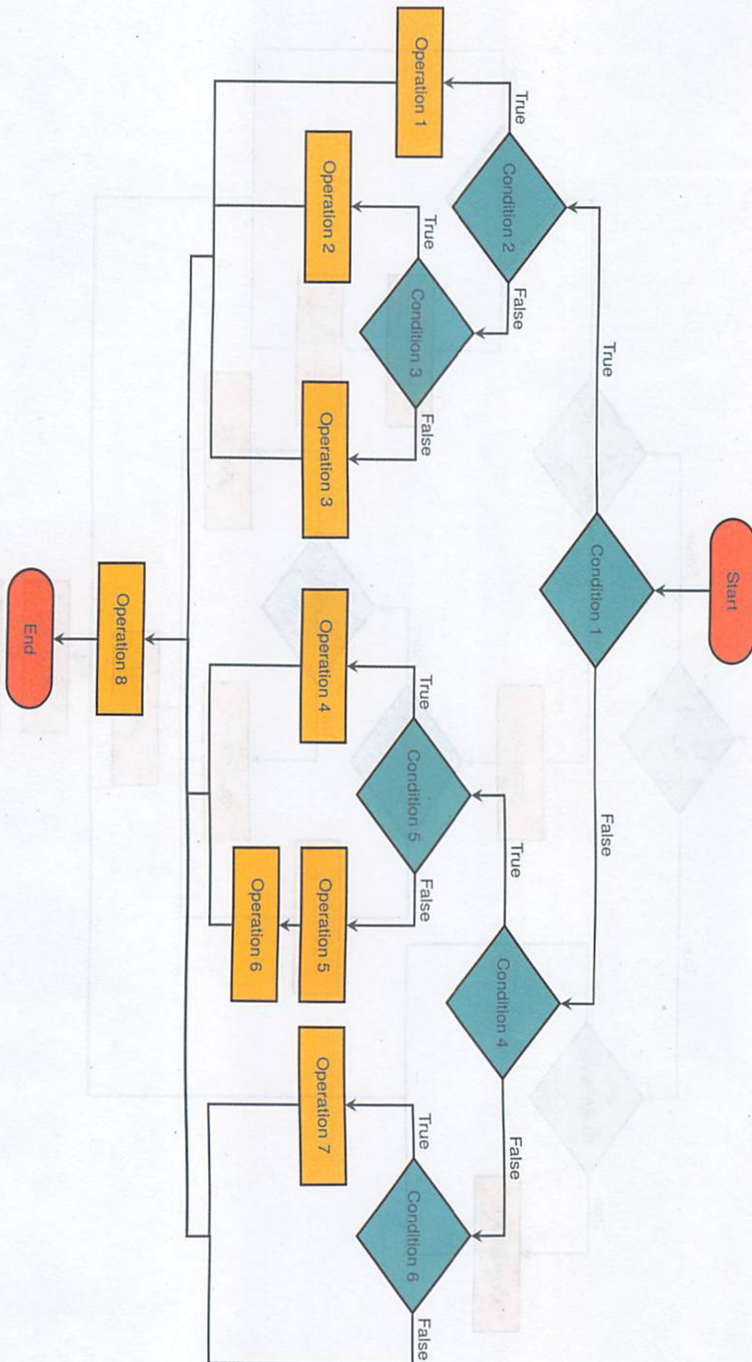
6

เฉลยข้อที่ 22.

6

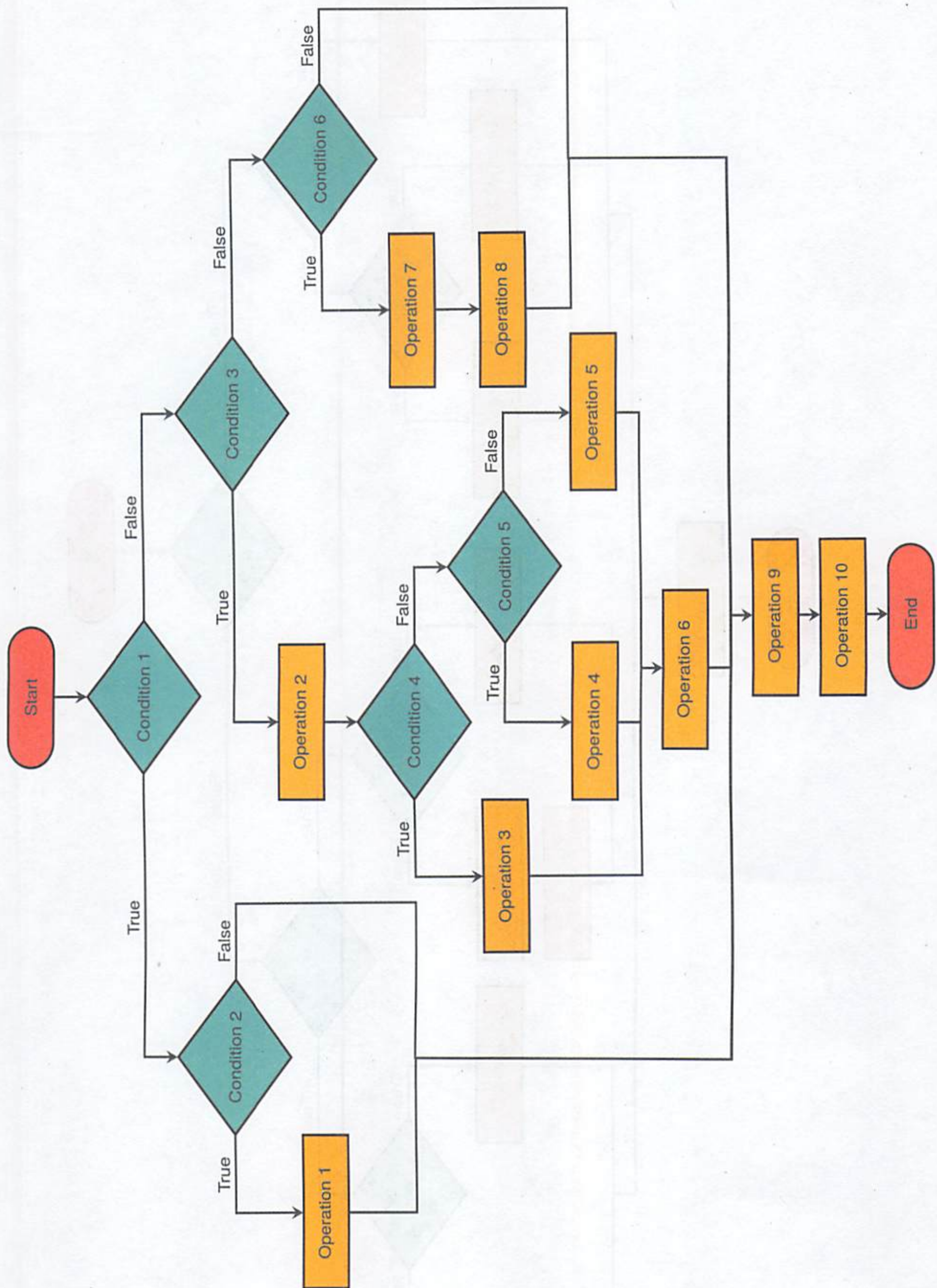

Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Operation 2
Operation 3
Condition 3
True
False
Operation 4
Condition 4
True
False
Operation 5
Condition 5
True
False
Operation 6
Operation 7
Operation 8
Operation 9
Condition 6
True
False
Operation 10
Condition 7
True
False
Operation 11
Operation 12
End

เฉลยข้อที่ 23.

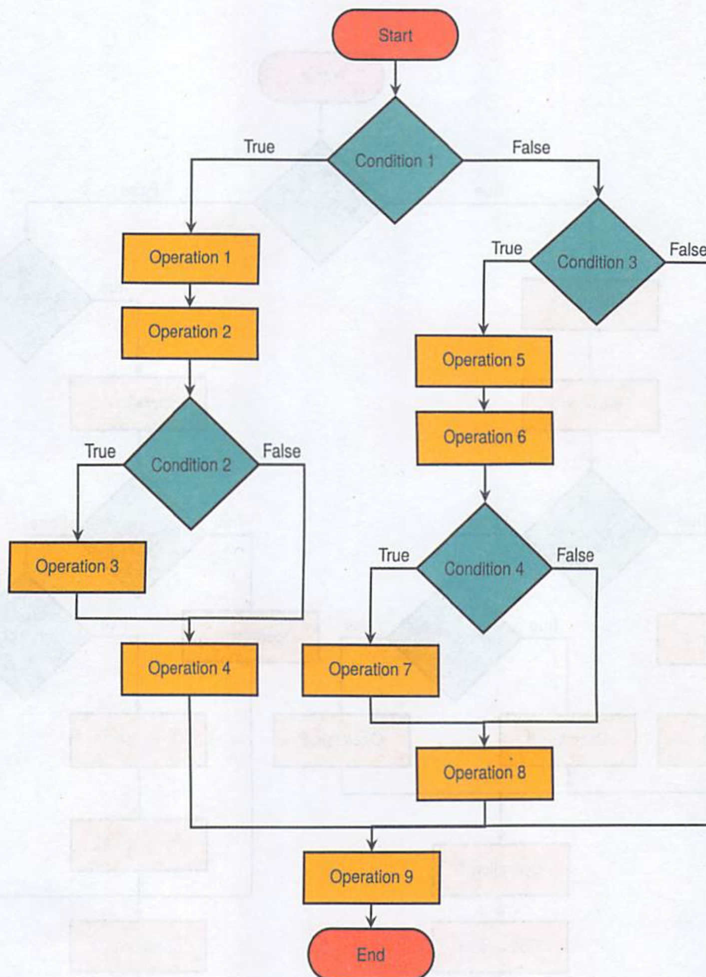

Start
Condition 1
True
False
Condition 2
True
False
Operation 1
Condition 3
True
False
Operation 2
Operation 3
Condition 4
True
False
Condition 5
True
False
Operation 4
Operation 5
Operation 6
Condition 6
True
False
Operation 7
Operation 8
End

6

เฉลยข้อที่ 24.

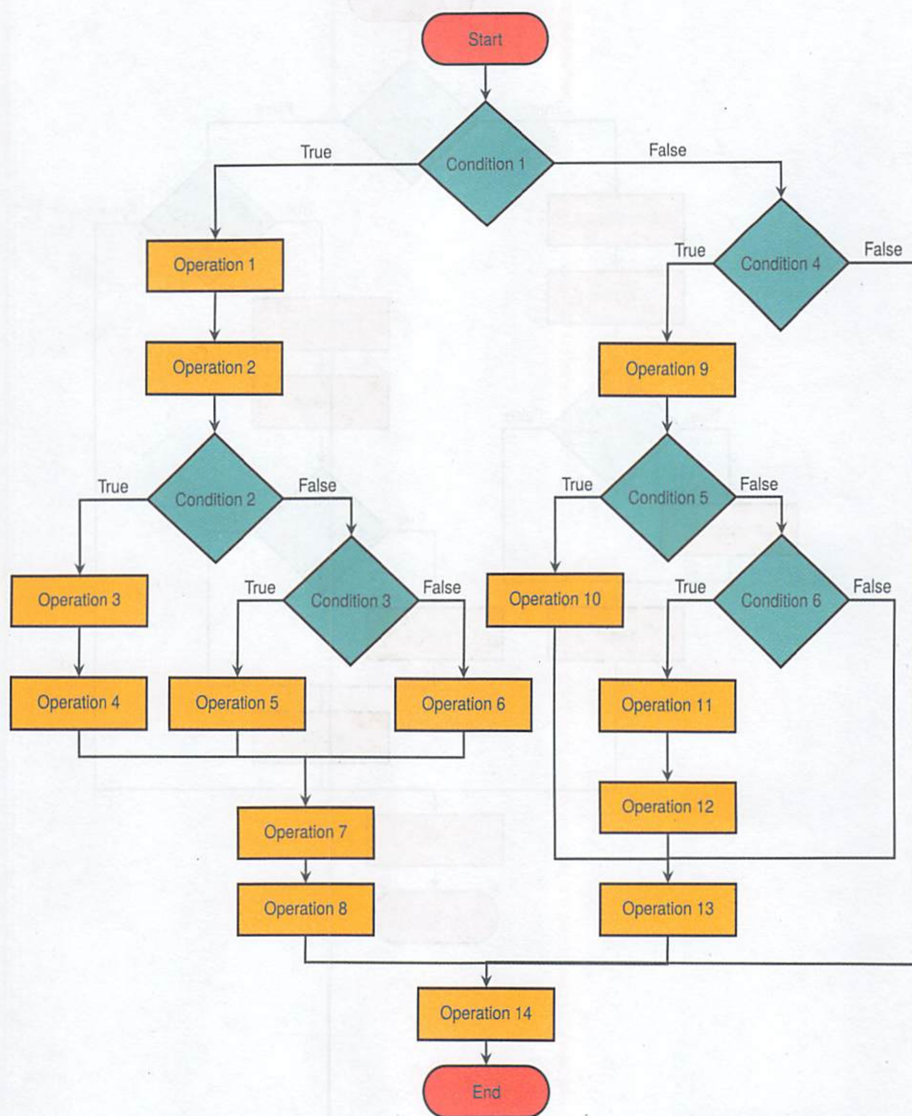

Start
Condition 1
True
False
Condition 2
True
False
Operation 1
Condition 3
True
False
Operation 2
Condition 4
True
False
Operation 3
Condition 5
True
False
Operation 4
Operation 5
Operation 6
Condition 6
True
False
Operation 7
Operation 8
Operation 9
Operation 10
End

เฉลยข้อที่ 25.

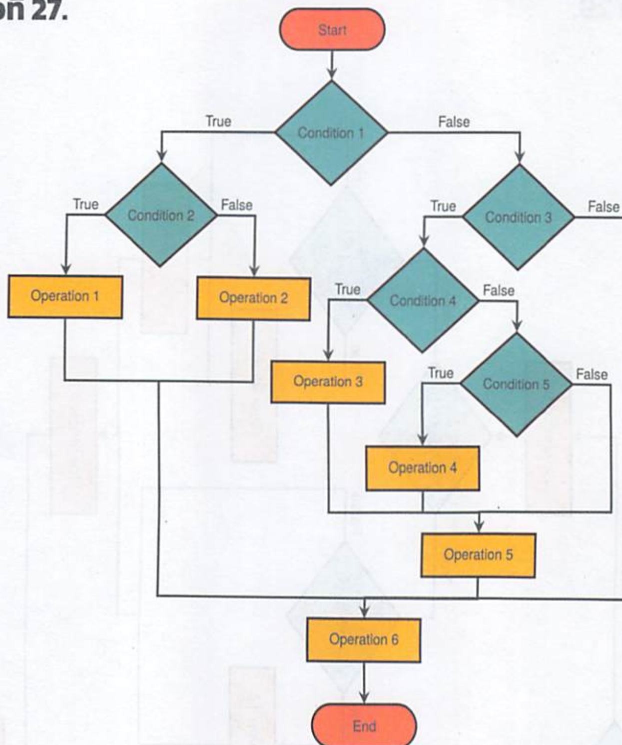

Start
Condition 1
True
False
Operation 1
Operation 2
Condition 2
True
False
Operation 3
Operation 4
Condition 3
True
False
Operation 5
Operation 6
Condition 4
True
False
Operation 7
Operation 8
Operation 9
End

6

เฉลยข้อที่ 26.

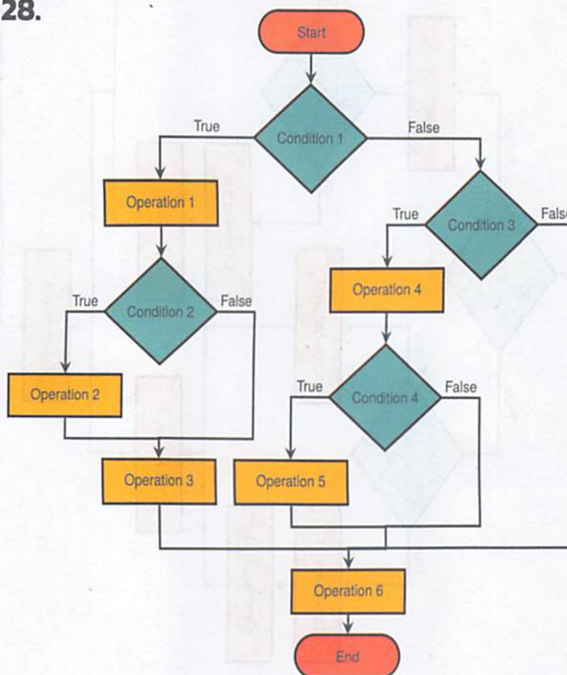

Start
Condition 1
True
False
Operation 1
Operation 2
Condition 4
True
False
Operation 9
Condition 2
True
False
Condition 3
True
False
Operation 3
Operation 4
Operation 5
Operation 6
Condition 5
True
False
Operation 10
Condition 6
True
False
Operation 11
Operation 12
Operation 7
Operation 8
Operation 13
Operation 14
End

เฉลยข้อที่ 27.



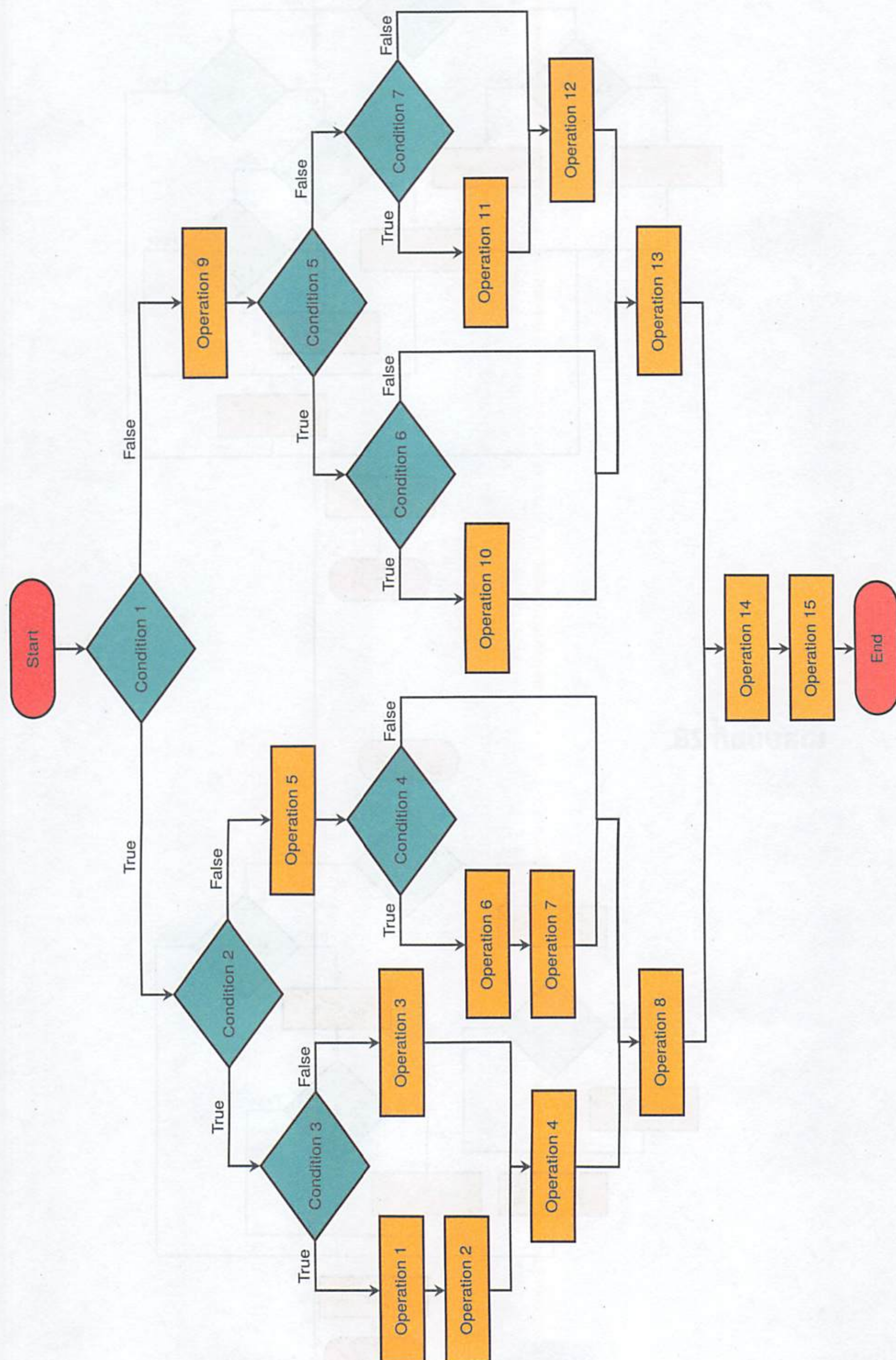
6

เฉลยข้อที่ 28.

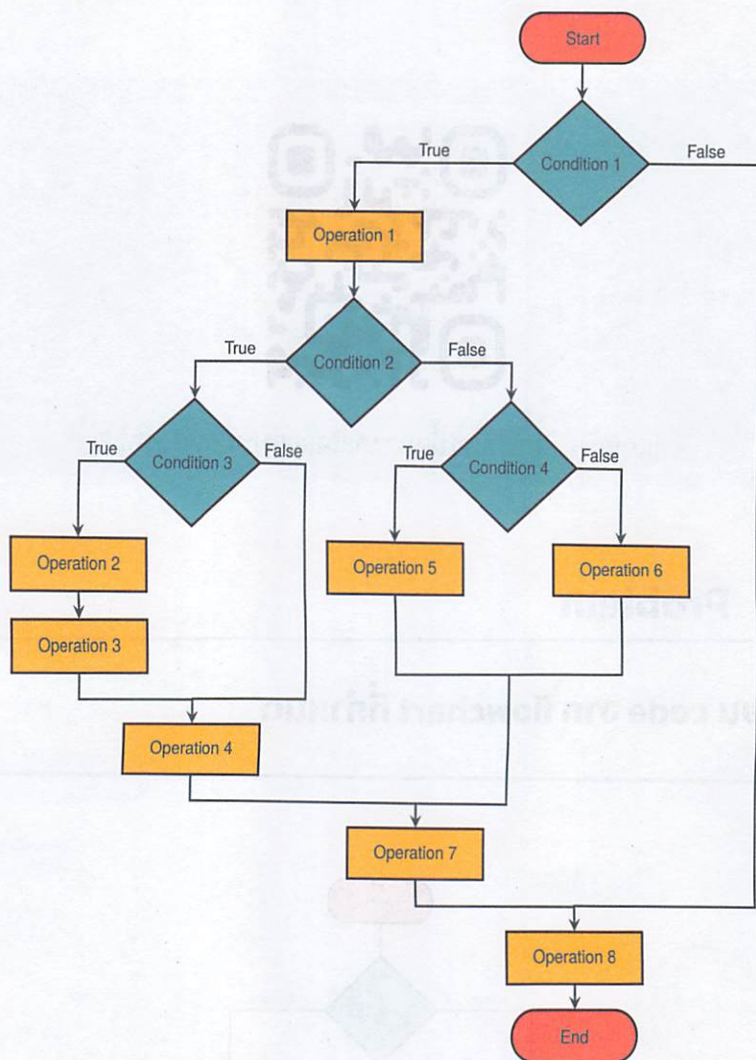


6

เฉลยข้อที่ 29.


Start
Condition 1
True
False
Condition 2
Operation 9
Condition 3
Operation 5
Condition 5
Condition 4
Condition 6
Condition 7
Operation 1
Operation 2
Operation 3
Operation 4
Operation 6
Operation 7
Operation 8
Operation 10
Operation 11
Operation 12
Operation 13
Operation 14
Operation 15
End

เฉลยข้อที่ 30.


Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Condition 3
True
False
Condition 4
True
False
Operation 2
Operation 3
Operation 4
Operation 5
Operation 6
Operation 7
Operation 8
End

6

07 IF-ELSE (CODE-FLOWCHART)

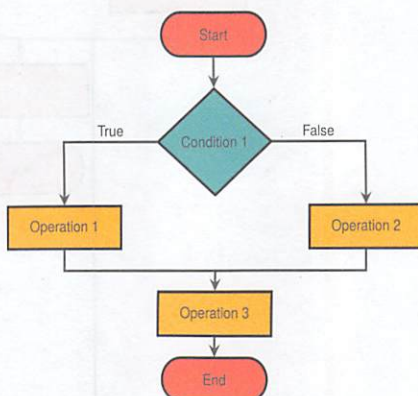


สแกนเพื่อเรียนวิดีโอเนื้อหาและเฉลยแบบฝึกหัดบทที่ 7

7.1 Problem

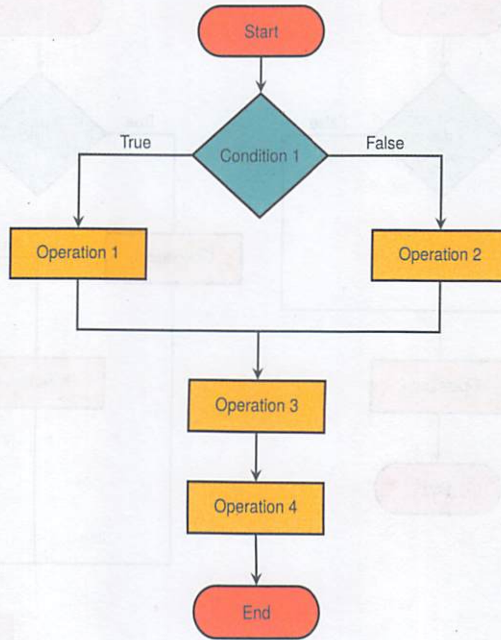
เขียน code จาก flowchart ที่กำหนด

Ex.

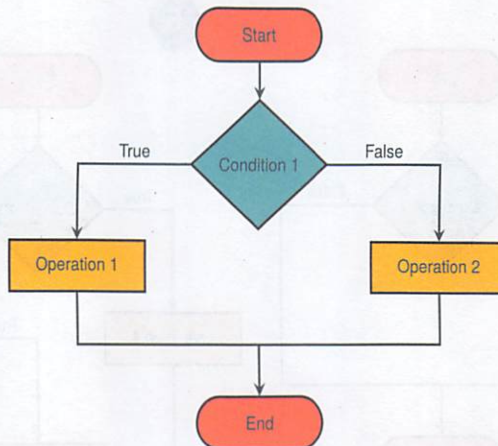



```
if condition1:
    operation1
else:
    operation2
operation3
```

1.

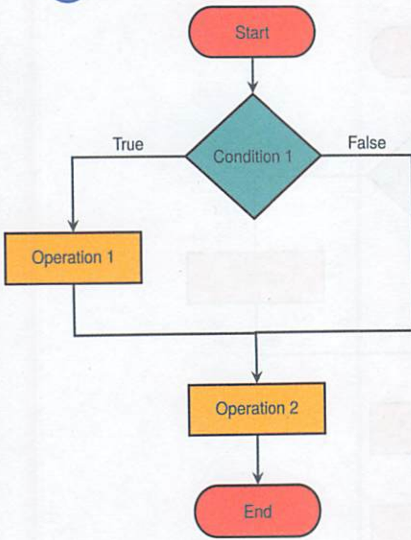

Start
True
Condition 1
False
Operation 1
Operation 2
Operation 3
Operation 4
End

2.

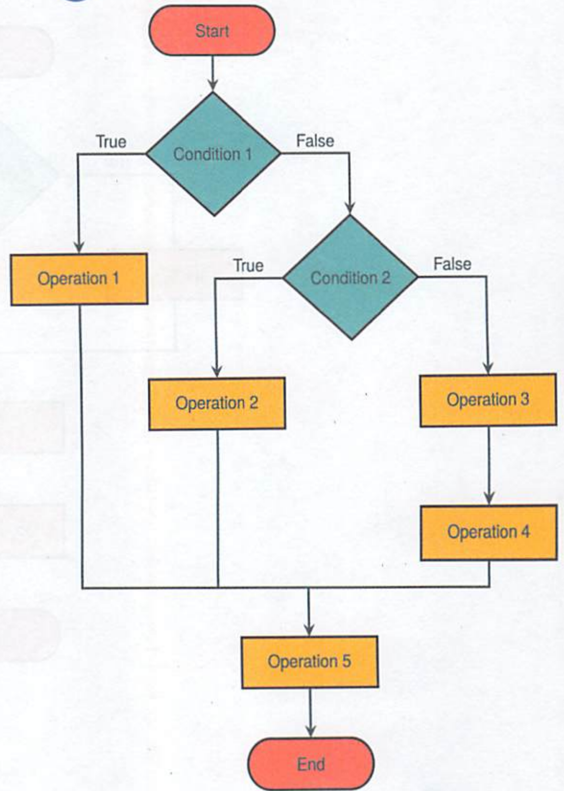

Start
True
Condition 1
False
Operation 1
Operation 2
End

7

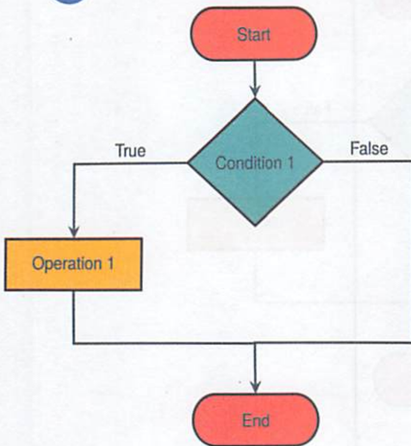
3.


Start
Condition 1
True
False
Operation 1
Operation 2
End

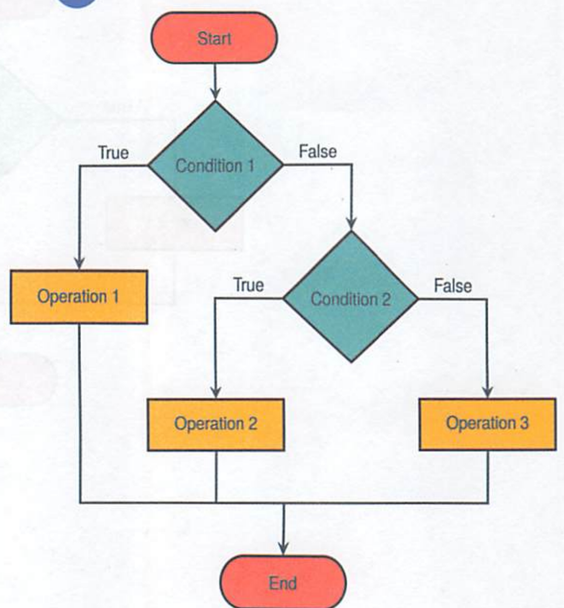
5.


Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Operation 2
Operation 3
Operation 4
Operation 5
End

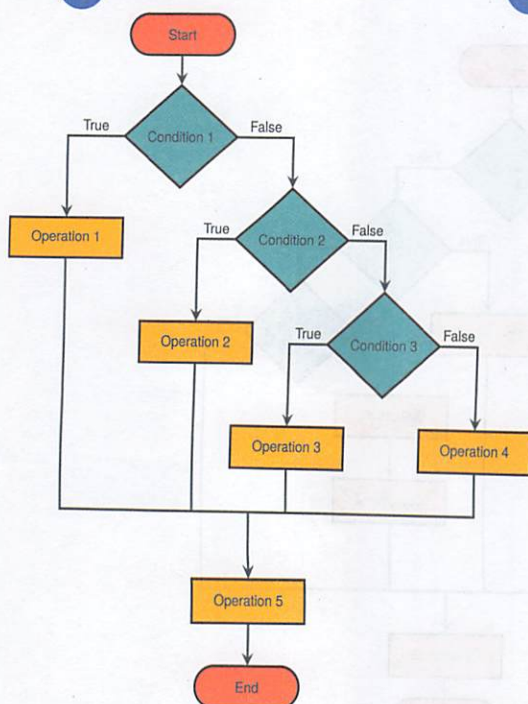
4.


Start
Condition 1
True
False
Operation 1
End

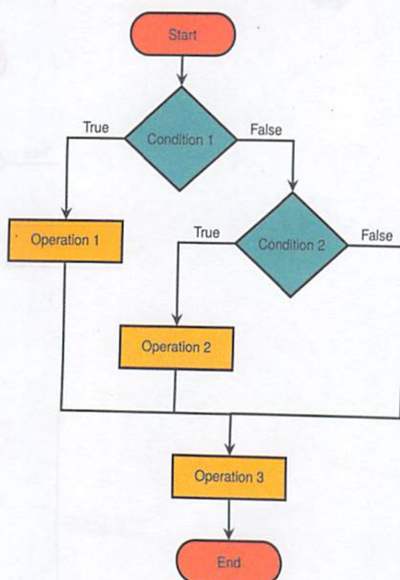
6.


Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Operation 2
Operation 3
End

7.

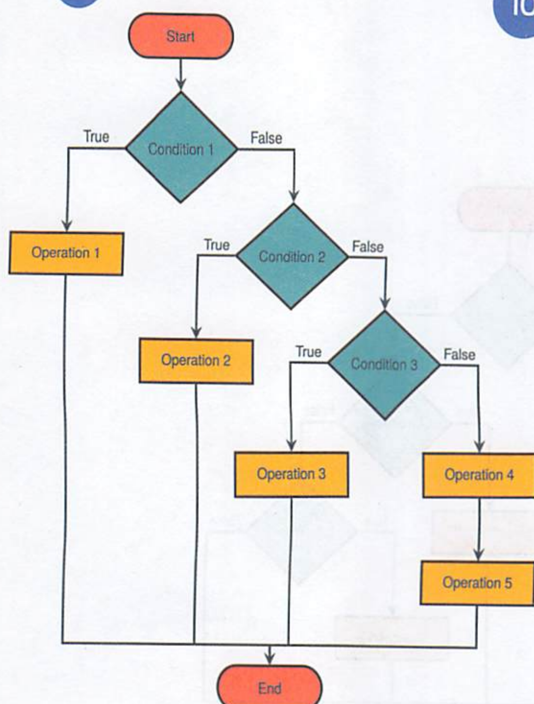

Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Operation 2
Condition 3
True
False
Operation 3
Operation 4
Operation 5
End

9.

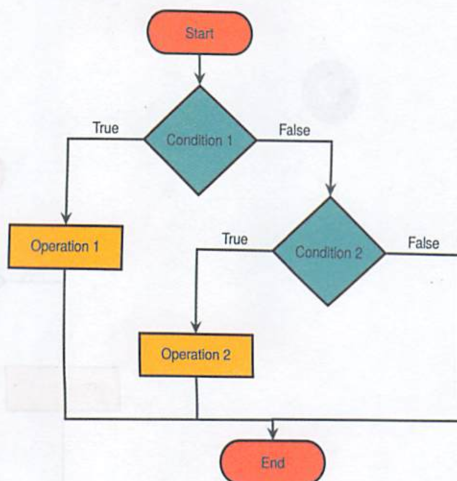

Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Operation 2
Operation 3
End

7

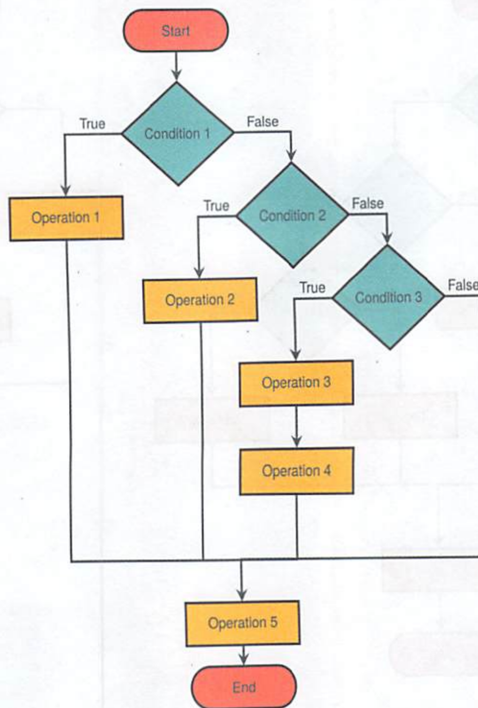
8.


Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Operation 2
Condition 3
True
False
Operation 3
Operation 4
Operation 5
End

10.

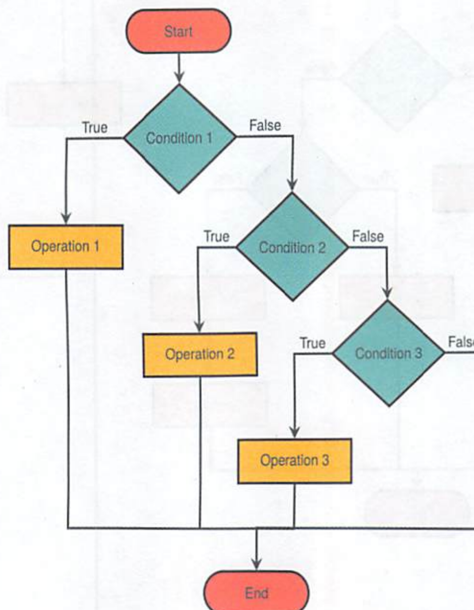

Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Operation 2
End

11.

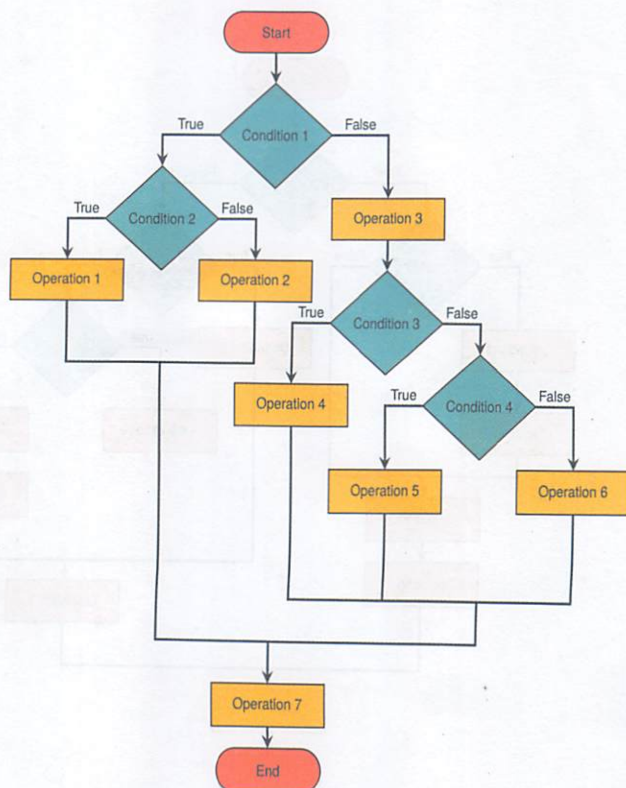

Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Operation 2
Condition 3
True
False
Operation 3
Operation 4
Operation 5
End

7

12.

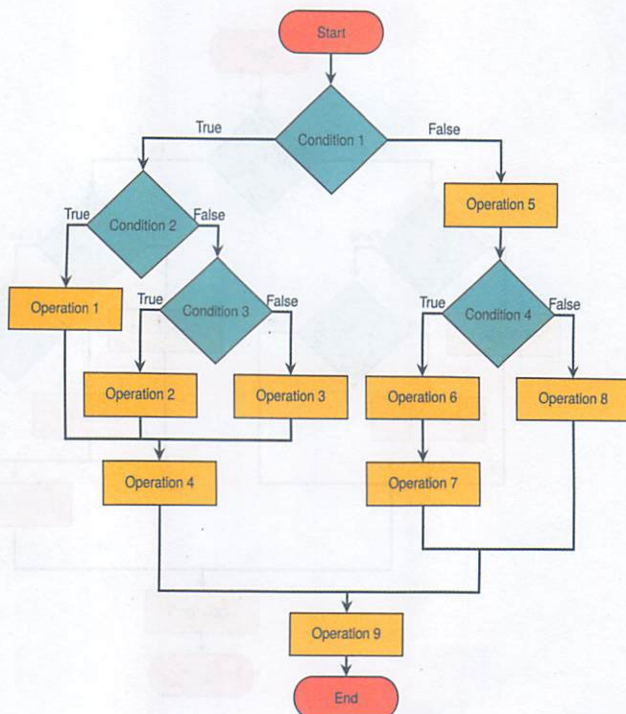

Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Operation 2
Condition 3
True
False
Operation 3
End

13.

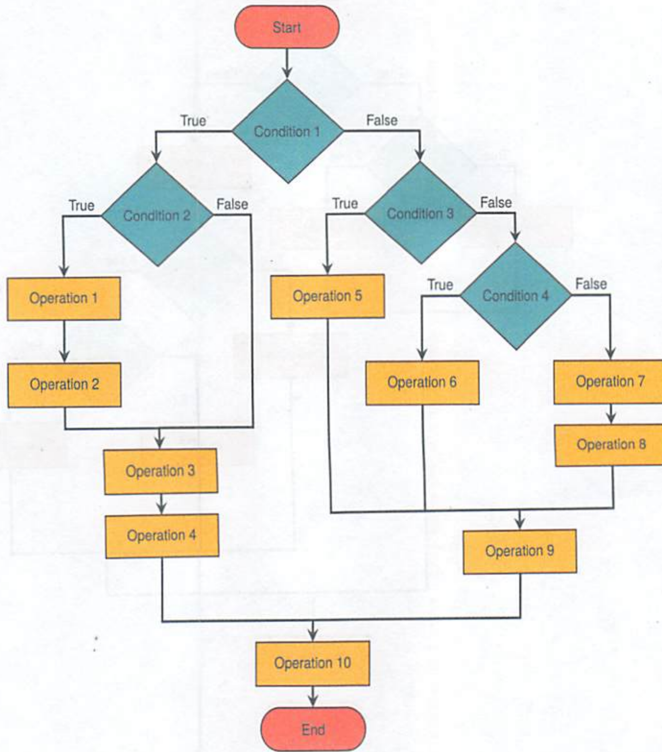

Start
Condition 1
True
False
Condition 2
True
False
Operation 3
Operation 1
Operation 2
Condition 3
True
False
Operation 4
Condition 4
True
False
Operation 5
Operation 6
Operation 7
End

7

14.

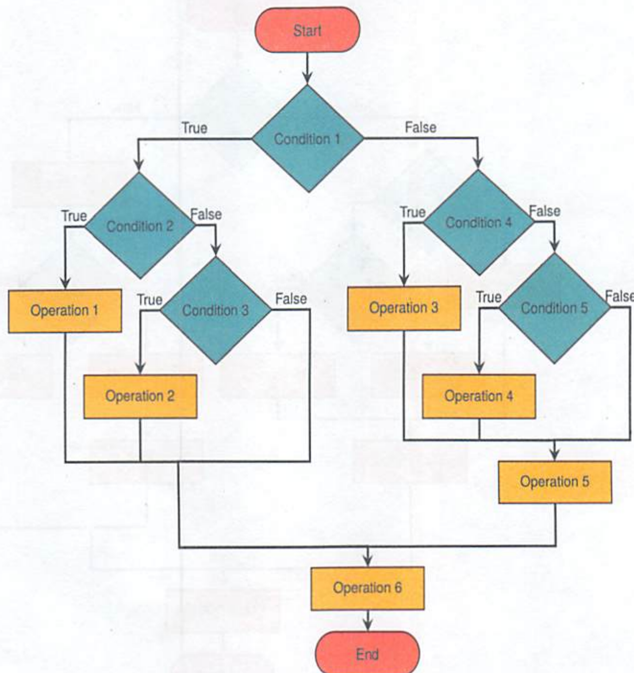

Start
Condition 1
True
False
Operation 5
Condition 2
True
False
Operation 1
Condition 3
True
False
Condition 4
True
False
Operation 2
Operation 3
Operation 6
Operation 8
Operation 4
Operation 7
Operation 9
End

15.

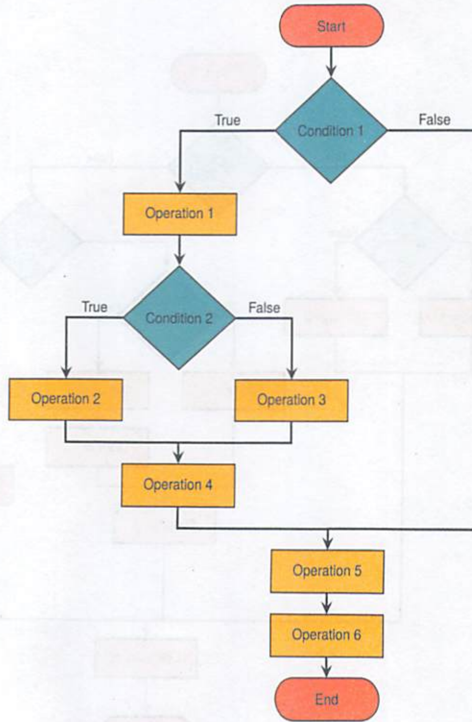

Start
Condition 1
True
False
Condition 2
True
False
Condition 3
True
False
Operation 1
Operation 2
Operation 3
Operation 4
Operation 5
Condition 4
True
False
Operation 6
Operation 7
Operation 8
Operation 9
Operation 10
End

7

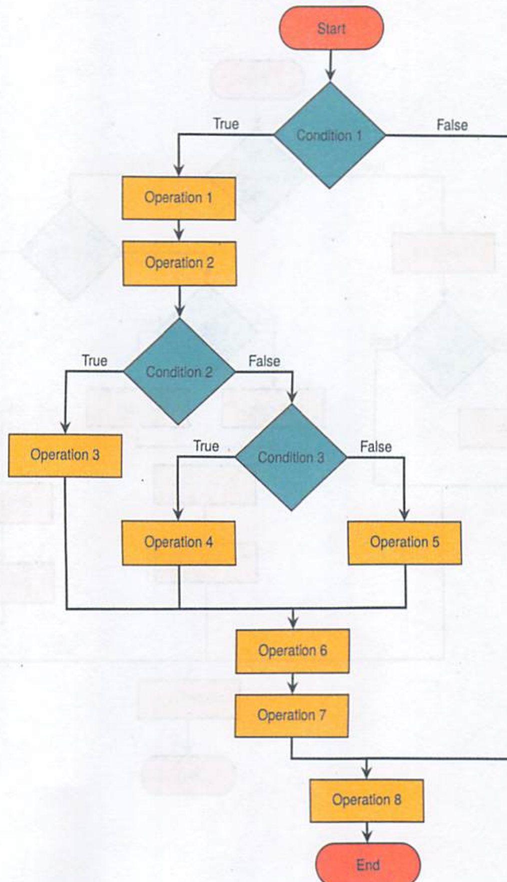
16.


Start
Condition 1
True
False
Condition 2
True
False
Condition 3
True
False
Condition 4
True
False
Condition 5
True
False
Operation 1
Operation 2
Operation 3
Operation 4
Operation 5
Operation 6
End

17.

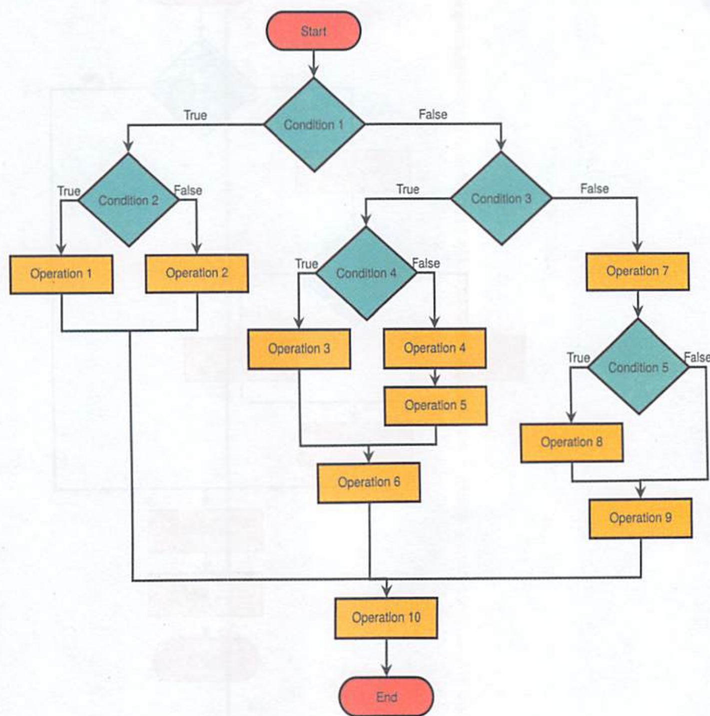

Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Operation 2
Operation 3
Operation 4
Operation 5
Operation 6
End

18.


Start
Condition 1
True
False
Operation 1
Operation 2
Condition 2
True
False
Operation 3
Condition 3
True
False
Operation 4
Operation 5
Operation 6
Operation 7
Operation 8
End

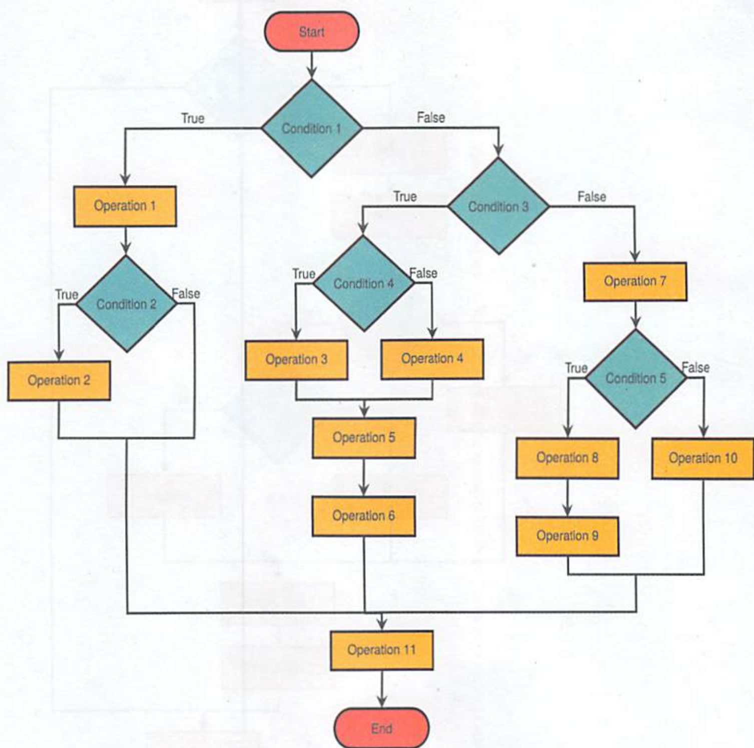
7

19.

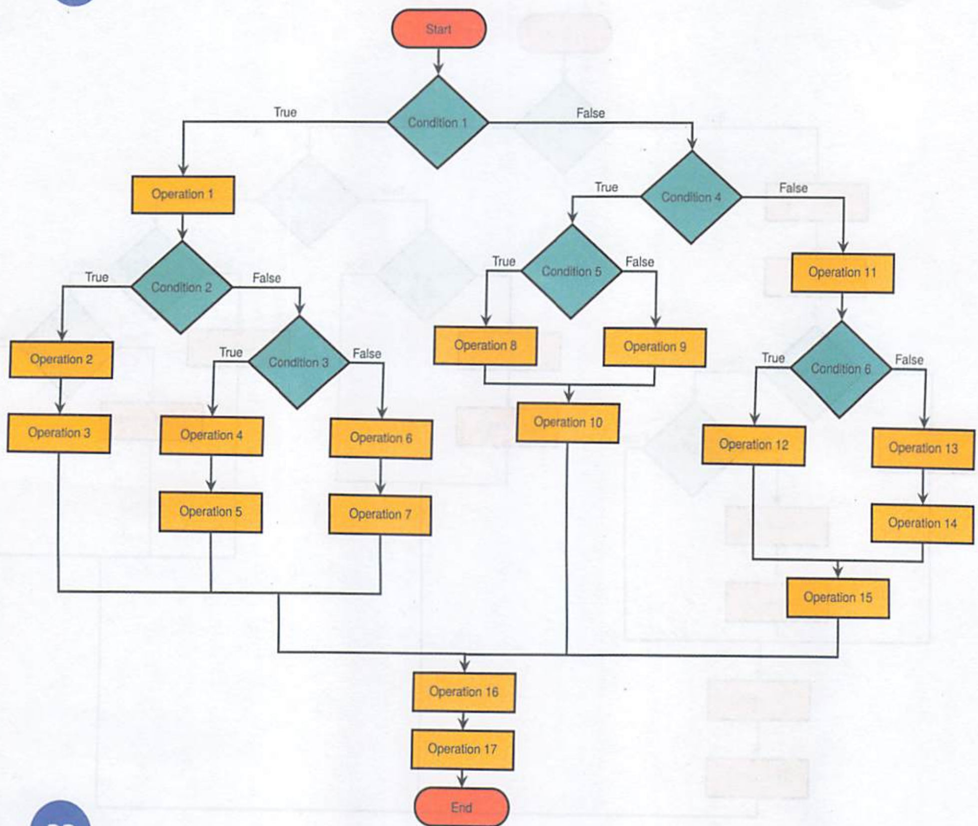

Start
Condition 1
True
False
Condition 2
True
False
Condition 3
True
False
Operation 1
Operation 2
Condition 4
True
False
Operation 7
Operation 3
Operation 4
Operation 5
Condition 5
True
False
Operation 8
Operation 6
Operation 9
Operation 10
End

7

20.

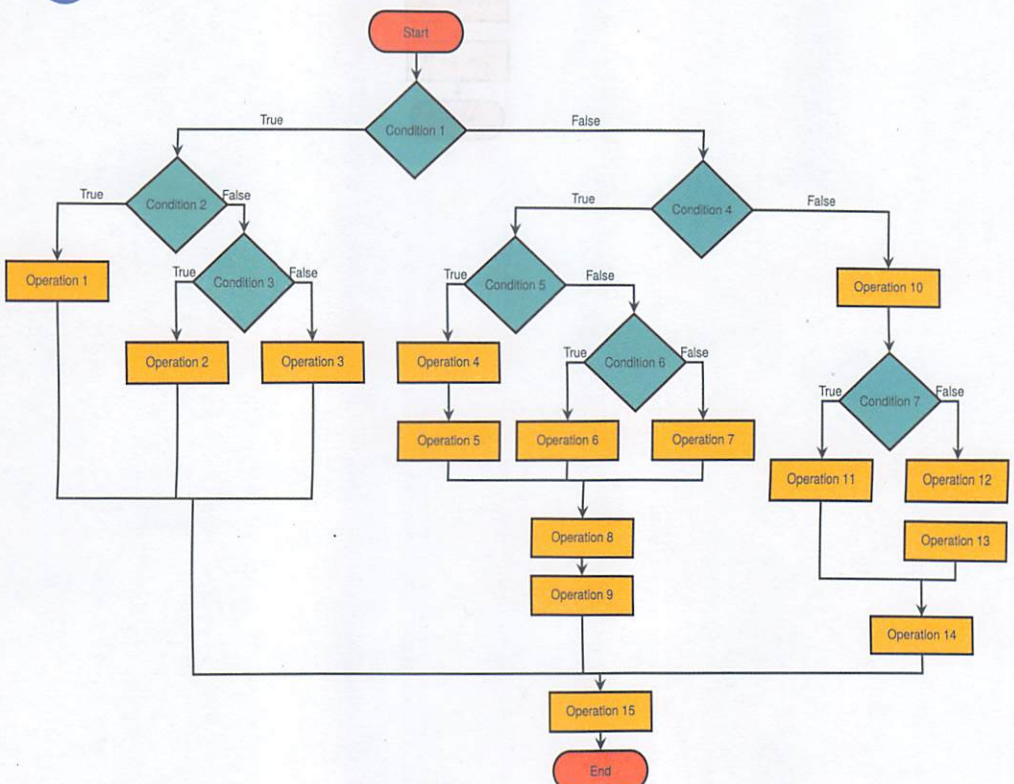

Start
Condition 1
True
False
Operation 1
Condition 3
True
False
Condition 2
True
False
Condition 4
True
False
Operation 7
Operation 2
Operation 3
Operation 4
Condition 5
True
False
Operation 5
Operation 8
Operation 10
Operation 6
Operation 9
Operation 11
End

21.

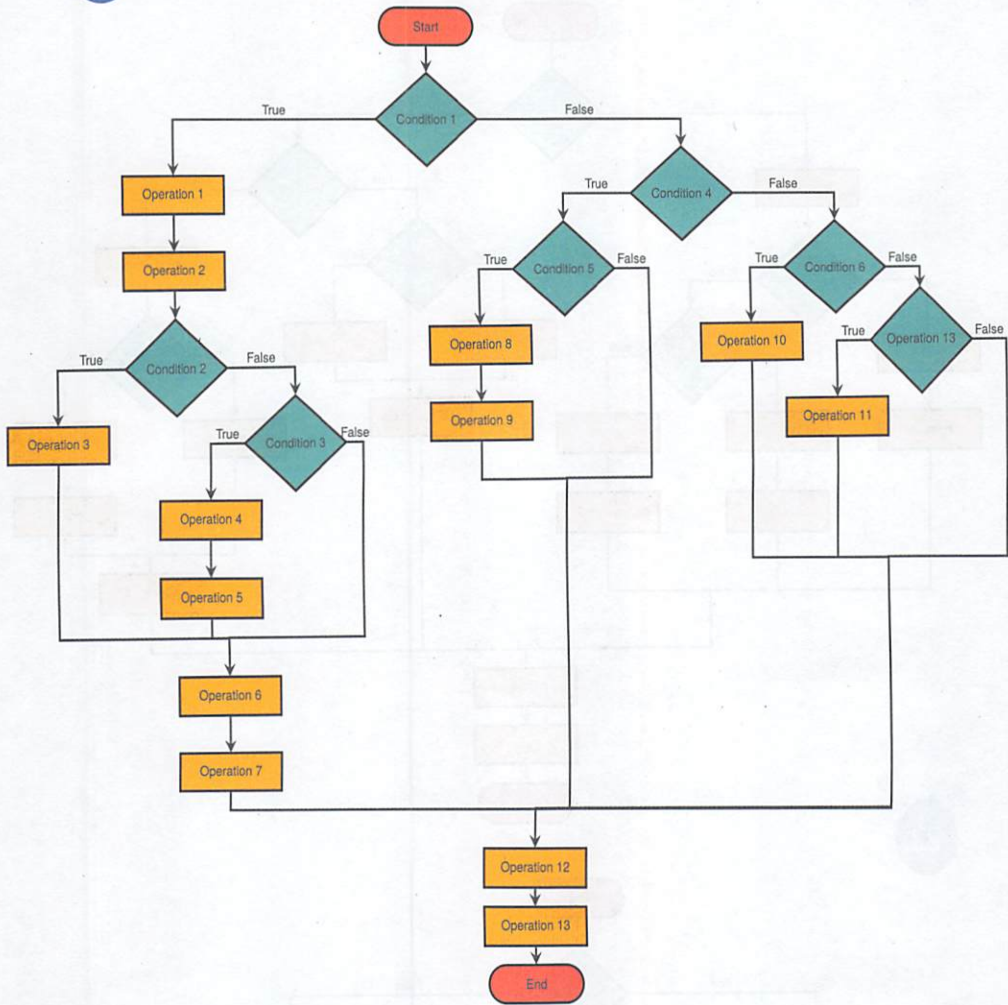

Start
Condition 1
True
False
Operation 1
Condition 2
True
False
Condition 4
True
False
Condition 5
True
False
Operation 11
Operation 2
Condition 3
True
False
Operation 8
Operation 9
Condition 6
True
False
Operation 3
Operation 4
Operation 6
Operation 10
Operation 12
Operation 13
Operation 5
Operation 7
Operation 14
Operation 15
Operation 16
Operation 17
End

7

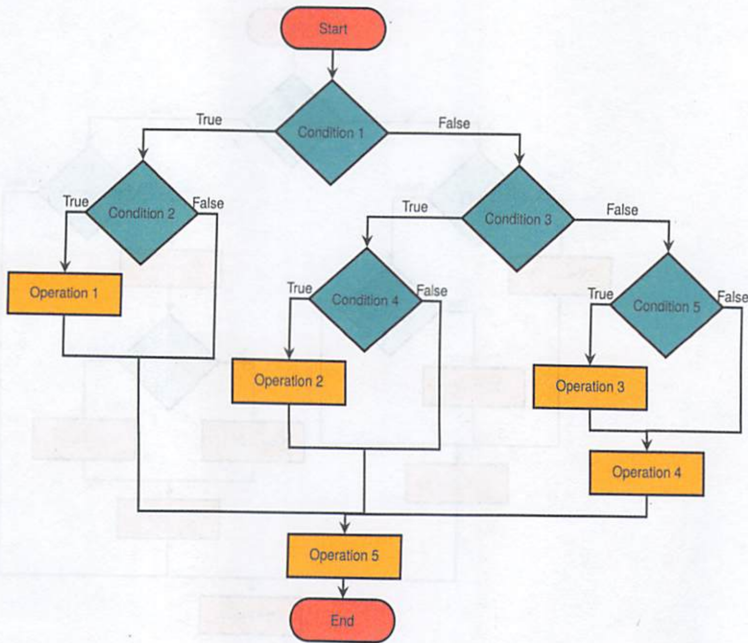
22.


Start
Condition 1
True
False
Condition 2
True
False
Condition 4
True
False
Operation 1
Condition 3
True
False
Condition 5
True
False
Operation 10
Operation 2
Operation 3
Operation 4
Condition 6
True
False
Condition 7
True
False
Operation 5
Operation 6
Operation 7
Operation 11
Operation 12
Operation 13
Operation 8
Operation 9
Operation 14
Operation 15
End

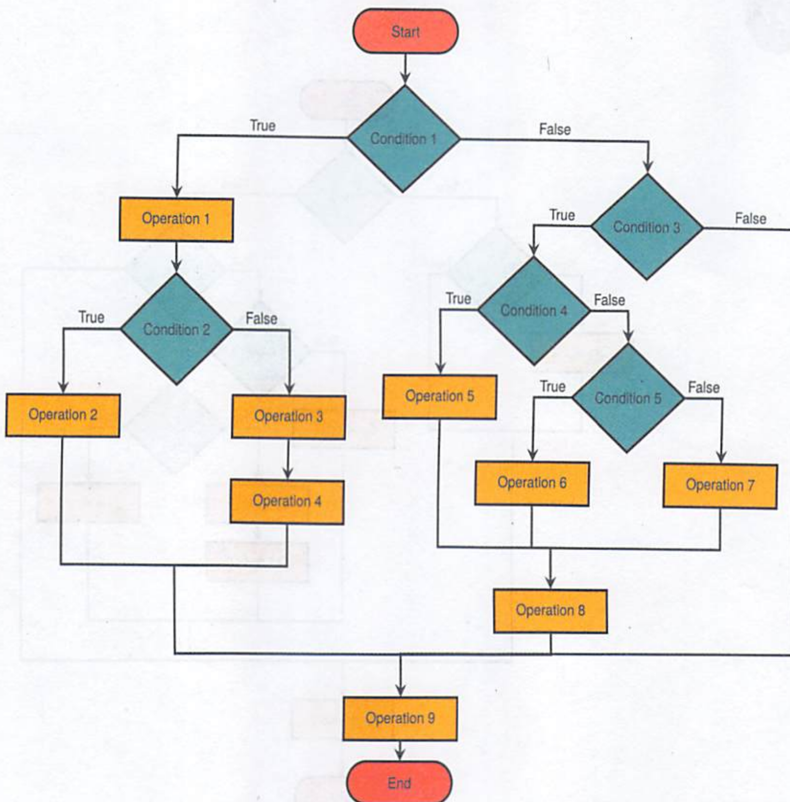
23.


Start
Condition 1
True
False
Operation 1
Operation 2
Condition 2
True
False
Operation 3
Condition 3
True
False
Operation 4
Operation 5
Operation 6
Operation 7
Condition 4
True
False
Condition 5
True
False
Operation 8
Operation 9
Condition 6
True
False
Operation 10
Operation 13
True
False
Operation 11
Operation 12
Operation 13
End

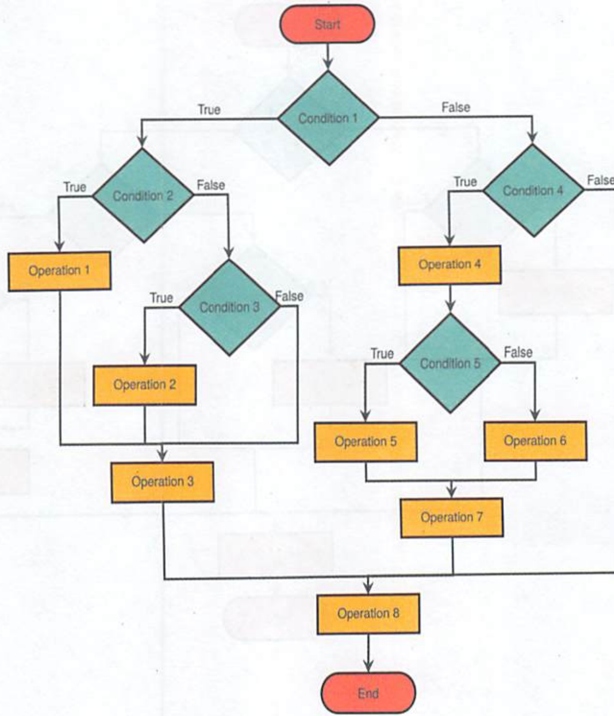
24.


Start
Condition 1
True
False
Condition 2
True
False
Condition 3
True
False
Operation 1
Condition 4
True
False
Condition 5
True
False
Operation 2
Operation 3
Operation 4
Operation 5
End

25.

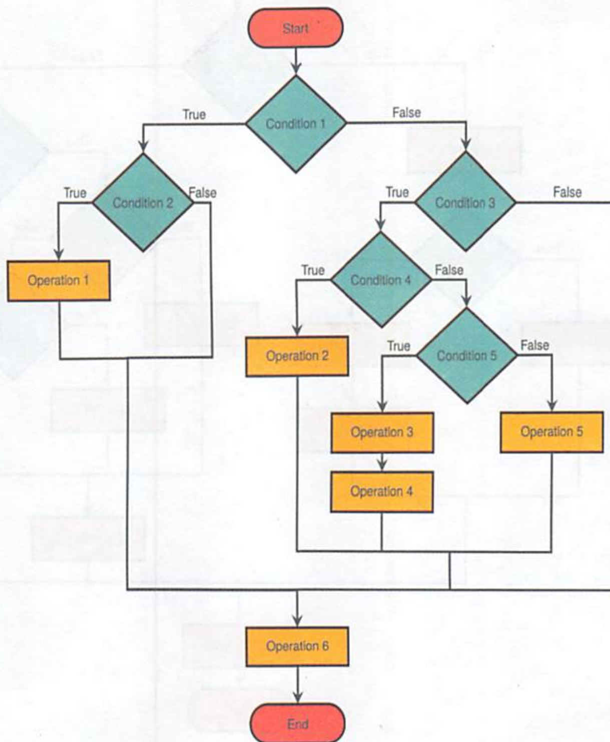

Start
Condition 1
True
False
Operation 1
Condition 3
True
False
Condition 2
True
False
Condition 4
True
False
Operation 2
Operation 3
Operation 5
Condition 5
True
False
Operation 4
Operation 6
Operation 7
Operation 8
Operation 9
End

26.

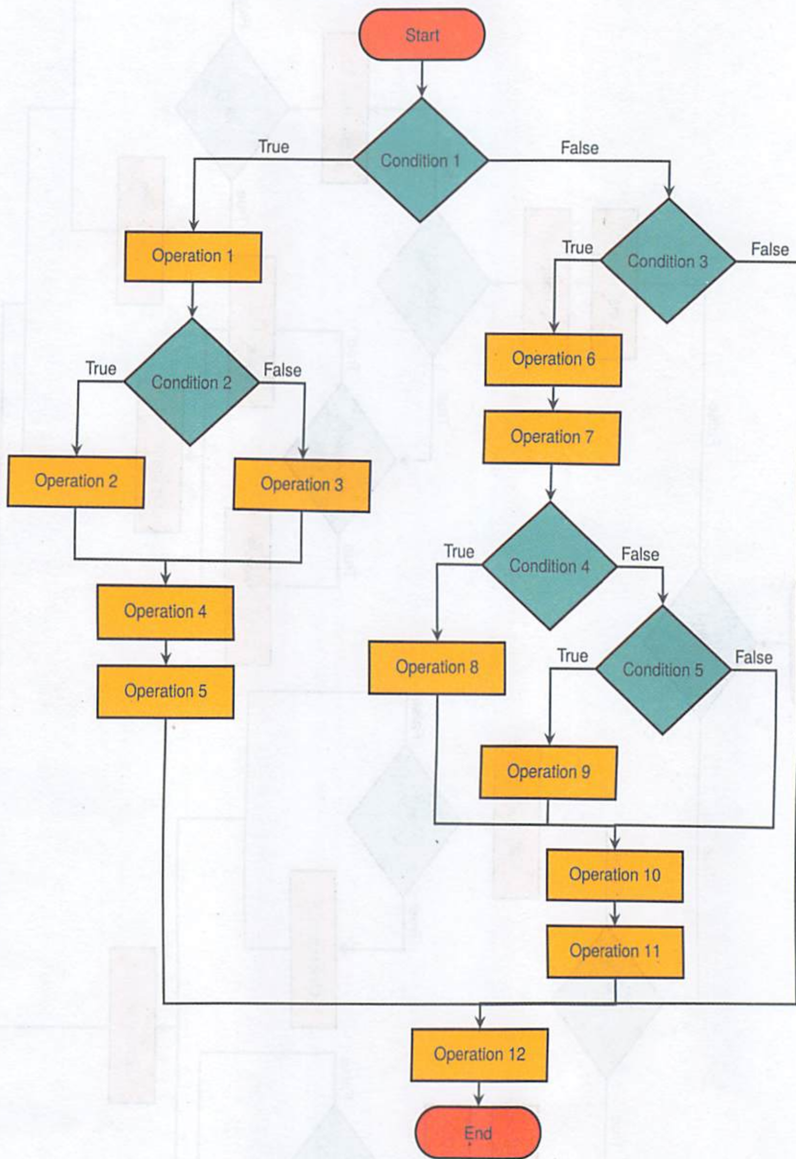

Start
Condition 1
True
False
Condition 2
True
False
Condition 4
True
False
Operation 1
Condition 3
True
False
Operation 4
Operation 2
Condition 5
True
False
Operation 5
Operation 6
Operation 3
Operation 7
Operation 8
End

7

27.


Start
Condition 1
True
False
Condition 2
True
False
Condition 3
True
False
Operation 1
Condition 4
True
False
Operation 2
Condition 5
True
False
Operation 3
Operation 5
Operation 4
Operation 6
End

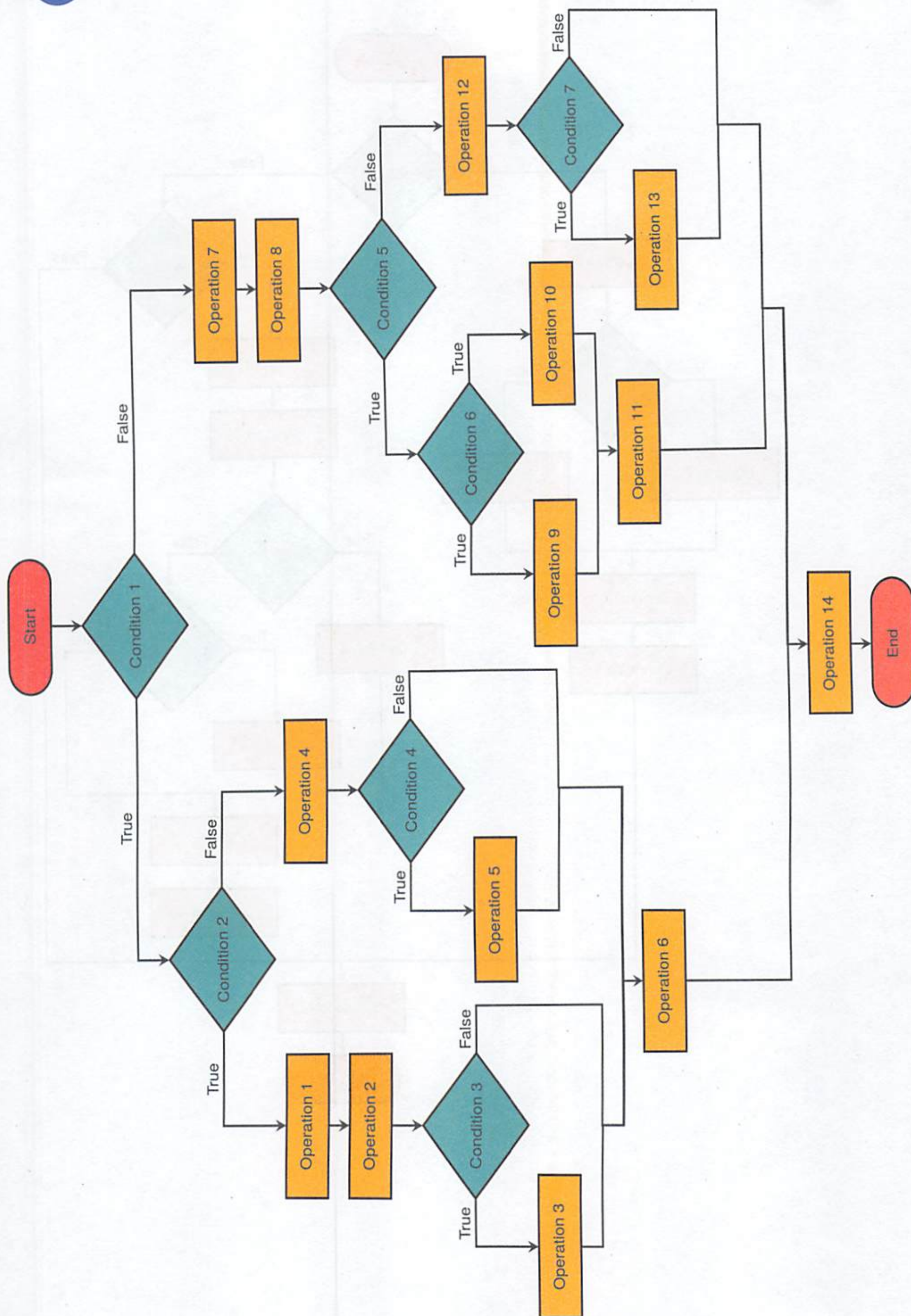
28.


Start
Condition 1
True
False
Operation 1
Condition 3
True
False
Condition 2
True
False
Operation 6
Operation 7
Operation 2
Operation 3
Condition 4
True
False
Operation 4
Operation 8
Condition 5
True
False
Operation 5
Operation 9
Operation 10
Operation 11
Operation 12
End

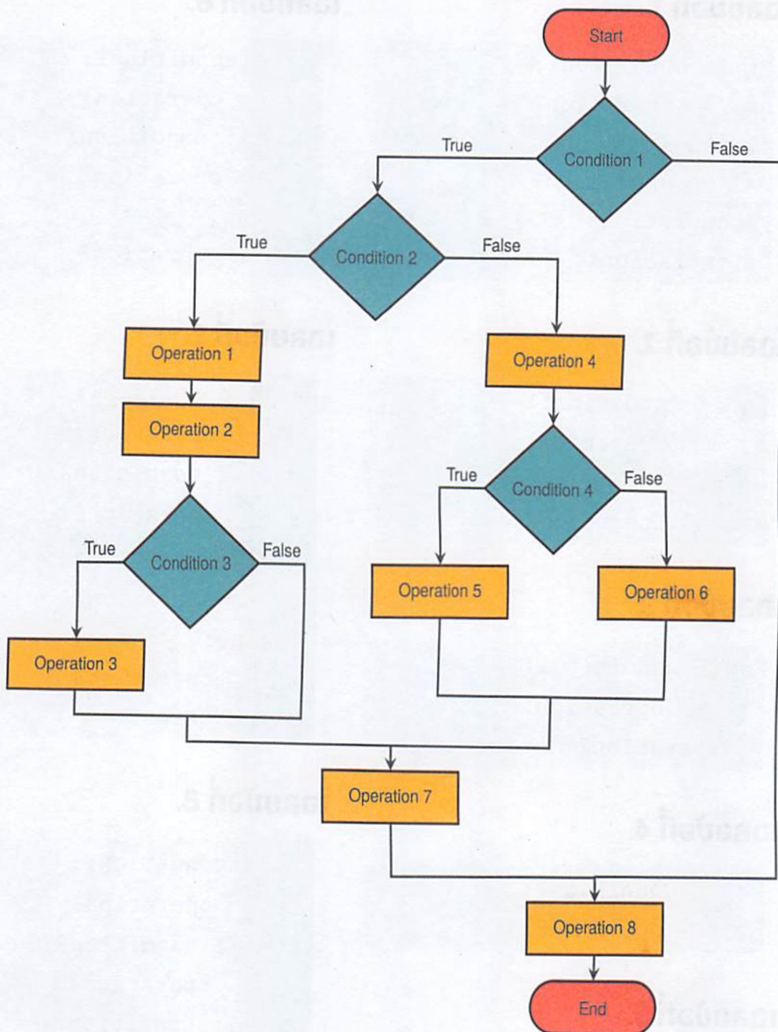
7

7

29.


Start
Condition 1
True
False
Condition 2
True
False
Operation 1
Operation 2
Condition 3
True
False
Operation 3
Operation 4
Condition 4
True
False
Operation 5
Operation 6
Operation 7
Operation 8
Condition 5
True
False
Condition 6
True
True
Operation 9
Operation 10
Operation 11
Operation 12
Condition 7
True
False
Operation 13
Operation 14
End

30.


Start
Condition 1
True
False
Condition 2
True
False
Operation 1
Operation 2
Operation 4
Condition 3
True
False
Condition 4
True
False
Operation 3
Operation 5
Operation 6
Operation 7
Operation 8
End

7

7.2 Solution

เฉลยข้อที่ 1.

```
if condition1:
    operation1
else:
    operation2
operation3
operation4
```

เฉลยข้อที่ 2.

```
if condition1:
    operation1
else:
    operation2
```

เฉลยข้อที่ 3.

```
if condition1:
    operation1
operation2
```

เฉลยข้อที่ 4.

```
if condition1:
    operation1
```

เฉลยข้อที่ 5.

```
if condition1:
    operation1
elif condition2:
    operation2
else:
    operation3
    operation4
operation5
```

เฉลยข้อที่ 6.

```
if condition1:
    operation1
elif condition2:
    operation2
else:
    operation3
```

เฉลยข้อที่ 7.

```
if condition1:
    operation1
elif condition2:
    operation2
elif condition3:
    operation3
else:
    operation4
operation5
```

เฉลยข้อที่ 8.

```
if condition1:
    operation1
elif condition2:
    operation2
elif condition3:
    operation3
else:
    operation4
    operation5
```

เฉลยข้อที่ 9.

```
if condition1:
    operation1
elif condition2:
    operation2
operation3
```

เฉลยข้อที่ 10.

```
if condition1:
    operation1
elif condition2:
    operation2
```

เฉลยข้อที่ 11.

```
if condition1:
    operation1
elif condition2:
    operation2
elif condition3:
    operation3
    operation4
operation5
```

เฉลยข้อที่ 12.

```
if condition1:
    operation1
elif condition2:
    operation2
elif condition3:
    operation3
```

เฉลยข้อที่ 13.

```
if condition1:
    if condition2:
        operation1
    else:
        operation2
else:
    operation3
    if condition3:
        operation4
    elif condition4:
        operation5
    else:
        operation6
operation7
```

เฉลยข้อที่ 14.

```
if condition1:
    if condition2:
        operation1
    elif condition3:
        operation2
    else:
        operation3
    operation4
else:
    operation5
    if condition4:
        operation6
        operation7
    else:
        operation8
operation9
```

เฉลยข้อที่ 15.

```
if condition1:
    if condition2:
        operation1
        operation2
    operation3
    operation4
else:
    if condition3:
        operation5
    elif condition4:
        operation6
    else:
        operation7
        operation8
    operation9
operation10
```

7

เฉลยข้อที่ 16.

```
if condition1:
    if condition2:
        operation1
    elif condition3:
        operation2
else:
    if condition4:
        operation3
    elif condition5:
        operation4
    operation5
operation6
```

เฉลยข้อที่ 17.

```
if condition1:
    operation1
    if condition2:
        operation2
    else:
        operation3
    operation4
operation5
operation6
```

เฉลยข้อที่ 18.

```
if condition1:
    operation1
    operation2
    if condition2:
        operation3
    elif condition3:
        operation4
    else:
        operation5
    operation6
    operation7
operation8
```

เฉลยข้อที่ 19.

```
if condition1:
    if condition2:
        operation1
    else:
        operation2
elif condition3:
    if condition4:
        operation3
    else:
        operation4
        operation5
    operation6
else:
    operation7
    if condition5:
        operation8
    operation9
operation10
```

เฉลยข้อที่ 20.

```
if condition1:
    operation1
    if condition2:
        operation2
elif condition3:
    if condition4:
        operation3
    else:
        operation4
    operation5
    operation6
else:
    operation7
    if condition5:
        operation8
        operation9
    else:
        operation10
operation11
```

เฉลยข้อที่ 21.

```
if condition1:
    operation1
    if condition2:
        operation2
        operation3
    elif condition3:
        operation4
        operation5
    else:
        operation6
        operation7
elif condition4:
    if condition5:
        operation8
    else:
        operation9
    operation10
else:
    operation11
    if condition6:
        operation12
    else:
        operation13
        operation14
    operation15
operation16
operation17
```

เฉลยข้อที่ 22.

```
if condition1:
    if condition2:
        operation1
    elif condition3:
        operation2
    else:
        operation3
elif condition4:
    if condition5:
        operation4
        operation5
    elif condition6:
        operation6
    else:
        operation7
    operation8
    operation9
else:
    operation10
    if condition7:
        operation11
    else:
        operation12
        operation13
    operation14
operation15
```

เฉลยข้อที่ 23.

```
if condition1:
    operation1
    operation2
    if condition2:
        operation3
    elif condition3:
        operation4
        operation5
    operation6
    operation7
elif condition4:
    if condition5:
        operation8
        operation9
else:
    if condition6:
        operation10
    elif condition7:
        operation11
operation12
operation13
```

เฉลยข้อที่ 24.

```
if condition1:
    if condition2:
        operation1
elif condition3:
    if condition4:
        operation2
else:
    if condition5:
        operation3
    operation4
operation5
```

เฉลยข้อที่ 25.

```
if condition1:
    operation1
    if condition2:
        operation2
    else:
        operation3
        operation4
elif condition3:
    if condition4:
        operation5
    elif condition5:
        operation6
    else:
        operation7
    operation8
operation9
```

เฉลยข้อที่ 26.

```
if condition1:
    if condition2:
        operation1
    elif condition3:
        operation2
    operation3
elif condition4:
    operation4
    if condition5:
        operation5
    else:
        operation6
    operation7
operation8
```

เฉลยข้อที่ 27.

```
if condition1:
    if condition2:
        operation1
elif condition3:
    if condition4:
        operation2
    elif condition5:
        operation3
        operation4
    else:
        operation5
operation6
```

เฉลยข้อที่ 28.

```
if condition1:
    operation1
    if condition2:
        operation2
    else:
        operation3
    operation4
    operation5
elif condition3:
    operation6
    operation7
    if condition4:
        operation8
    elif condition5:
        operation9
    operation10
    operation11
operation12
```

เฉลยข้อที่ 29.

```
if condition1:
    if condition2:
        operation1
        operation2
        if condition3:
            operation3
    else:
        operation4
        if condition4:
            operation5
    operation6
else:
    operation7
    operation8
    if condition5:
        if condition6:
            operation9
        else:
            operation10
        operation11
    else:
        operation12
        if condition7:
            operation13
operation14
```

เฉลยข้อที่ 30.

```
if condition1:
    if condition2:
        operation1
        operation2
        if condition3:
            operation3
    else:
        operation4
        if condition4:
            operation5
        else:
            operation6
    operation7
operation8
```

08 IF-ELSE (PROBLEM)


สแกนเพื่อเรียนวิดีโอเนื้อหา
และเฉลยแบบฝึกหัดบทที่ 8


สแกนเพื่อรับ code บทที่ 8

8.1 Problem

จงวาด flowchart และเขียน code ของโจทย์ต่อไปนี้

1. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม และตรวจสอบว่าจำนวนที่รับมาเป็นเลขคู่หรือเลขคี่
    - ถ้าเป็นเลขคู่ให้พิมพ์ 'เลขคู่'
    - ถ้าเป็นเลขคี่ให้พิมพ์ 'เลขคี่'
2. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม และตรวจสอบว่าจำนวนที่รับมามีค่ามากกว่า 0 หรือไม่
    - ถ้ามีค่ามากกว่า 0 ให้พิมพ์ 'มากกว่า 0'
    - ถ้ามีค่าน้อยกว่าหรือเท่ากับ 0 ให้พิมพ์ 'น้อยกว่าหรือเท่ากับ 0'
3. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนจริง และตรวจสอบว่าจำนวนที่รับมามีค่ามากกว่า 0, น้อยกว่า 0 หรือเท่ากับ 0
    - ถ้ามีค่ามากกว่า 0 ให้พิมพ์ 'positive'
    - ถ้ามีค่าน้อยกว่า 0 ให้พิมพ์ 'negative'
    - ถ้ามีค่าเท่ากับ 0 ให้พิมพ์ 'zero'

4. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนจริง และตรวจสอบว่าจำนวนที่รับมามีค่ามากกว่า 0, น้อยกว่า 0 หรือเท่ากับ 0

- ถ้ามีค่ามากกว่า 0 ให้ตรวจสอบต่อว่าเป็นเลขคู่หรือเลขคี่
  - ถ้าเป็นเลขคู่ให้พิมพ์ 'positive even'
  - ถ้าเป็นเลขคี่ให้พิมพ์ 'positive odd'
- ถ้ามีค่าน้อยกว่า 0 ให้ตรวจสอบต่อว่าเป็นเลขคู่หรือเลขคี่
  - ถ้าเป็นเลขคู่ให้พิมพ์ 'negative even'
  - ถ้าเป็นเลขคี่ให้พิมพ์ 'negative odd'
- ถ้ามีค่าเท่ากับ 0 ให้พิมพ์ 'zero'

5. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นจำนวนเต็ม (width, length) และตรวจสอบว่า ถ้าอินพุตตัวใดตัวหนึ่งหรือทั้งสองตัวเป็นศูนย์หรือจำนวนเต็มลบให้พิมพ์ 'กรุณากรอกจำนวนเต็มบวก' แต่ถ้าเป็นจำนวนเต็มบวกทั้งคู่ให้คำนวณพื้นที่สี่เหลี่ยมและพิมพ์ออกมา

8

6. เขียนโปรแกรมรับอินพุต 1 ตัวเป็นปีคริสตศักราช (จำนวนเต็ม) และตรวจสอบว่า จำนวนที่รับมามีค่ามากกว่าหรือเท่ากับ 0 หรือไม่

- ถ้ามีค่ามากกว่าหรือเท่ากับ 0 ให้คำนวณปีพุทธศักราช (ปีคริสตศักราช + 543) และพิมพ์ค่าออกมา
- ถ้ามีค่าน้อยกว่า 0 ให้พิมพ์ 'กรุณากรอกข้อมูลที่มากกว่าหรือเท่ากับ 0'

7. เขียนโปรแกรมรับอินพุต 1 ตัวเป็นอุณหภูมิในหน่วยฟาเรนไฮต์ (จำนวนจริง) และตรวจสอบว่า จำนวนที่รับมามีค่ามากกว่าหรือเท่ากับ 32 หรือไม่

- ถ้ามีค่ามากกว่าหรือเท่ากับ 32 ให้คำนวนอุณหภูมิในหน่วยองศาเซลเซียส และพิมพ์ค่าออกมา

$$C = \frac{(5 \times (F - 32))}{9}$$

- ถ้ามีค่าน้อยกว่า 32 ให้พิมพ์ 'cold'

8. เขียนโปรแกรมรับอินพุต 1 ตัวเป็นจำนวนเงินในหน่วยดอลล่าสหรัฐ (จำนวนจริง) และตรวจสอบว่า จำนวนที่รับมามีค่ามากกว่า 0 หรือไม่
   - ถ้ามีค่ามากกว่า 0 ให้คำนวณจำนวนเงินในหน่วยบาท (THB = USDx32.5) และพิมพ์ค่าออกมา
   - ถ้ามีค่าน้อยกว่าหรือเท่ากับ 0 ให้พิมพ์ "you don't have money"

9. เขียนโปรแกรมรับอินพุต 1 ตัว เป็นจำนวนเงินในหน่วยบาท (จำนวนจริง) และตรวจสอบว่า จำนวนที่รับมามีค่ามากกว่า 0 หรือไม่
   - ถ้ามีค่ามากกว่า 0 ให้คำนวณจำนวนเงินในหน่วยดอลล่าที่สามารถแลกได้ และกำไรที่ธนาคารได้ โดยที่ธนาคารขายเงินดอลล่าที่ราคา 32.80 บาท (ซึ่งธนาคารจะได้กำไร 0.30 บาท) พิมพ์จำนวนเงินดอลล่าที่แลกได้และกำไรของธนาคาร
   - ถ้ามีค่าน้อยกว่าหรือเท่ากับ 0 ให้พิมพ์ "you don't have money"

8

10. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นจำนวนจริง และตรวจสอบว่าจำนวนที่รับมาตัวใดมีค่ามากกว่า และพิมพ์จำนวนนั้นออกมา แต่ถ้าจำนวนทั้งสองมีค่าเท่ากันให้พิมพ์ 'มีค่าเท่ากัน'

11. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นสายอักขระ จากนั้นตรวจสอบว่า สายอักขระทั้ง 2 มีความยาวเท่ากันหรือไม่ ถ้ามีความยาวเท่ากันให้พิมพ์ 'same' ถ้าไม่เท่ากันให้พิมพ์ 'not same'

12. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม และตรวจสอบว่าจำนวนที่รับมาหารด้วย 3 ลงตัวหรือไม่ ถ้าลงตัวให้พิมพ์ 'หารด้วย 3 ลงตัว' แต่ถ้าไม่ลงตัวให้พิมพ์ 'หารด้วย 3 ไม่ลงตัว'

13. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม และตรวจสอบว่าจำนวนที่รับมาหารด้วย 3 และ 5 ลงตัวหรือไม่ ถ้าลงตัวให้พิมพ์ 'หารด้วย 3 และ 5 ลงตัว' แต่ถ้าไม่ลงตัวให้พิมพ์ 'หารด้วย 3 และ 5 ไม่ลงตัว'

14. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม และตรวจสอบว่าจำนวนที่รับมาหารด้วย 3 หรือ 5 ลงตัวหรือไม่ ถ้าลงตัวให้พิมพ์ 'หารด้วย 3 หรือ 5 ลงตัว' แต่ถ้าไม่ลงตัวให้พิมพ์ 'หารด้วย 3 หรือ 5 ไม่ลงตัว'

15. เขียนโปรแกรมรับอินพุต 3 ตัวที่เป็นจำนวนจริง และตรวจสอบว่าผลบวกของจำนวนที่หนึ่งและจำนวนที่สองมากกว่าจำนวนที่สามหรือไม่

- ถ้ามากกว่าให้พิมพ์ 'a + b > c'
- ถ้าไม่มากกว่า ไม่ดำเนินการใด ๆ

16. คุณเป็นโปรแกรมเมอร์ที่ห้างสรรพสินค้าชื่อดังแห่งหนึ่ง คุณได้รับมอบหมายให้เขียนโปรแกรมซึ่งมีรายละเอียดดังนี้

- รับอินพุต 2 ตัวที่เป็นจำนวนเต็ม (hour, minute) เพื่อคำนวณค่าที่จอดรถ
- โดยชั่วโมงแรกจอดฟรี หลังจากนั้นเสียค่าจอดชั่วโมงละ 30 บาท
- เศษของชั่วโมงนับเป็น 1 ชั่วโมง
- จากนั้นให้พิมพ์ค่าจอดรถออกมา

17. คุณเป็นโปรแกรมเมอร์ที่ห้างสรรพสินค้าชื่อดังแห่งหนึ่ง คุณได้รับมอบหมายให้เขียนโปรแกรมซึ่งมีรายละเอียดดังนี้

- รับอินพุต 2 ตัวที่เป็นจำนวนเต็ม (hour, minute) เพื่อคำนวณค่าจอดรถ
- โดยชั่วโมงแรกจอดฟรี หลังจากนั้นเสียค่าจอดชั่วโมงละ 30 บาท
- เศษของชั่วโมงนับเป็น 1 ชั่วโมง
- จากนั้นให้พิมพ์ค่าจอดรถออกมา
- แต่ถ้าผู้ใช้โปรแกรมป้อนอินพุตตัวใดตัวหนึ่ง หรือทั้งสองตัวเป็นจำนวนเต็มลบ ให้พิมพ์ 'โปรดใส่ข้อมูลที่ไม่ติดลบ' และไม่ต้องคำนวณค่าจอดรถ

18. คุณเป็นโปรแกรมเมอร์ที่ห้างสรรพสินค้าชื่อดังแห่งหนึ่ง คุณได้รับมอบหมายให้เขียนโปรแกรมซึ่งมีรายละเอียดดังนี้

- รับอินพุต 3 ตัวที่เป็นจำนวนเต็ม (bill, hour, minute) เพื่อคำนวณค่าจอดรถ โดยมีเงื่อนไขดังนี้
- ถ้า bill ≥ 1000 จะจอดรถฟรี 4 ชั่วโมง หลังจากนั้นจะเสียค่าจอดรถชั่วโมงละ 30 บาท
- ถ้า bill < 1000 จะจอดรถฟรี 1 ชั่วโมง หลังจากนั้นจะเสียค่าจอดชั่วโมงละ 30 บาท
- เศษของชั่วโมงนับเป็น 1 ชั่วโมง
- เมื่อคำนวณค่าจอดรถเสร็จให้ทำการพิมพ์ออกมา

8

19. คุณเป็นโปรแกรมเมอร์ที่ร้านขายสินค้าแบรนด์เนมชื่อดังแห่งหนึ่ง คุณได้รับมอบหมายให้เขียนโปรแกรมซึ่งมีรายละเอียดดังนี้

- รับอินพุต 1 ตัวที่เป็นจำนวนจริง (bill) เพื่อคำนวณยอดเงินที่ต้องจ่ายจากการซื้อสินค้า โดยที่ร้านค้ามีเงื่อนไขดังนี้
- ถ้าซื้อสินค้าราคารวมกันตั้งแต่ 1,000 บาทขึ้นไปจะลดราคาให้ 10%
- ถ้าซื้อสินค้าราคารวมกันตั้งแต่ 10,000 บาทขึ้นไปจะลดราคาให้ 15%
- ถ้าซื้อสินค้าราคารวมกันตั้งแต่ 50,000 บาทขึ้นไปจะลดราคาให้ 20%
- จงคำนวณยอดเงินที่ต้องจ่ายจากการซื้อสินค้าและพิมพ์ออกมา

20. คุณเป็นฟิตเนสเทรนเนอร์ในยิมชื่อดังแห่งหนึ่ง ในแต่ละวันมีคนมาขอคำปรึกษาเบื้องต้นเกี่ยวกับการดูแลรูปร่างกับคุณมากมาย คุณหมดเวลาไปกับการให้คำปรึกษาและรู้สึกเหนื่อย โชคดีที่คุณมีความรู้ในการเขียนโปรแกรม คุณจึงวางแผนที่จะสร้างระบบอัตโนมัติเพื่อให้คำปรึกษาเบื้องต้นกับลูกค้าที่มาใช้บริการยิม ซึ่งมีรายละเอียดดังนี้

8

- รับอินพุต 3 ตัวที่เป็นสายอักขระ, จำนวนจริง, และจำนวนเต็ม (gender, weight, height) เพื่อให้คำแนะนำสุขภาพโดยมีเงื่อนไขดังนี้
- ถ้าเป็นเพศชาย ให้พิจารณาว่าน้ำหนัก (กิโลกรัม) มากกว่าส่วนสูง (เซนติเมตร) - 100 หรือไม่
  - ถ้ามากกว่าให้พิมพ์ 'ควรออกกำลังกาย'
  - ถ้าไม่มากกว่าให้พิมพ์ 'คุณผู้ชายหุ่นดีเยี่ยม'
- ถ้าเป็นเพศหญิง ให้พิจารณาว่าน้ำหนัก (กิโลกรัม) มากกว่าส่วนสูง (เซนติเมตร) - 110 หรือไม่
  - ถ้ามากกว่าให้พิมพ์ 'ควรออกกำลังกาย'
  - ถ้าไม่มากกว่าให้พิมพ์ 'คุณผู้หญิงหุ่นดีเยี่ยม'

21. คุณเป็นตำรวจกรมทางหลวง หน้าที่ของคุณคือลงโทษผู้ที่ขับรถเร็วเกินอัตราที่กำหนด ณ ป้อมที่คุณอยู่มีเครื่องตรวจจับความเร็วอยู่แล้ว งานที่คุณต้องทำคือดูอัตราเร็วจากเครื่องตรวจจับและพิจารณาว่าต้องออกใบสั่งหรือไม่ ปัญหาที่คุณเผชิญคือมีจำนวนรถที่ขับผ่านป้อมของคุณเยอะมากจนทำให้คุณไม่สามารถทำงานได้ทัน โชคดีที่คุณมีความรู้ในการเขียนโปรแกรม คุณจึงวางแผนที่จะสร้างระบบอัตโนมัติขึ้นมา ซึ่งมีรายละเอียดดังนี้

- รับอินพุต 1 ตัวที่เป็นจำนวนจริง (velocity) เพื่อพิจารณาโทษจากความเร็วที่ขับโดยมีเงื่อนไขดังนี้

- ถ้าความเร็ว (velocity) มากกว่า 120 กิโลเมตร/ชั่วโมง ให้พิมพ์ 'ออกใบสั่ง'
- แต่ถ้าไม่ขับเร็วกว่า 120 กิโลเมตร/ชั่วโมง จะไม่ดำเนินการใด ๆ

22. คุณเป็นตำรวจกรมทางหลวง หน้าที่ของคุณคือลงโทษผู้ที่ขับรถเร็วเกินอัตราที่กำหนด ณ ป้อมที่คุณอยู่มีเครื่องตรวจจับความเร็วอยู่แล้ว งานที่คุณต้องทำคือดูอัตราเร็วจากเครื่องตรวจจับและพิจารณาโทษปรับ ปัญหาที่คุณเผชิญคือมีจำนวนรถที่ขับผ่านป้อมของคุณเยอะมากจนทำให้คุณไม่สามารถทำงานได้ทัน โชคดีที่คุณมีความรู้ในการเขียนโปรแกรม คุณจึงวางแผนที่จะสร้างระบบอัตโนมัติขึ้นมา ซึ่งมีรายละเอียดดังนี้

- รับอินพุต 1 ตัวที่เป็นจำนวนจริง (velocity) เพื่อพิจารณาโทษปรับจากความเร็วที่ขับโดยมีเงื่อนไขดังนี้
- ถ้าความเร็ว (velocity) มากกว่า 120 กิโลเมตร/ชั่วโมง ให้พิมพ์ 'ปรับ 1,000 บาท'
- ถ้าความเร็ว (velocity) มากกว่า 80 กิโลเมตร/ชั่วโมง แต่ไม่เกิน 120 กิโลเมตร/ชั่วโมง ให้พิมพ์ 'ปรับ 500 บาท'
- นอกนั้นจะไม่ดำเนินการใด ๆ

23. คุณเป็นโปรแกรมเมอร์ในบริษัทสร้างแอปพลิเคชั่นสำหรับฟังเพลงแห่งหนึ่ง บริษัทแห่งนี้ต้องการแจ้งเตือนผู้ใช้เมื่อมีการฟังเพลงต่อเนื่องเป็นเวลานานมากเกินไป คุณได้รับมอบหมายให้เขียนโปรแกรมซึ่งมีรายละเอียดดังนี้

- รับอินพุต 1 ตัวที่เป็นจำนวนจริง (listen_time) เพื่อแจ้งเตือนผู้ใช้เมื่อมีการฟังเพลงต่อเนื่องเป็นเวลานานเกินไป โดยมีเงื่อนไขดังนี้
- ถ้าใช้บริการ (listen_time) มากกว่า 4 ชั่วโมง ให้ทำการแจ้งเตือนว่า 'อันตรายต่อหู'
- ถ้ายังใช้ไม่เกิน 4 ชั่วโมง ให้ทำการแจ้งเตือนว่า 'ขอให้มีความสุขกับการฟังเพลง'

24. คุณเป็นโปรแกรมเมอร์ในบริษัทสร้างแอปพลิเคชั่นสำหรับการเดินทางชื่อดังแห่งหนึ่ง คุณได้รับมอบหมายให้เขียนโปรแกรมซึ่งมีรายละเอียดดังนี้

- รับอินพุต 2 ตัวที่เป็นสายอักขระ และจำนวนจริง (status, distance) เพื่อคำนวณค่าเดินทางในการใช้บริการ โดยมีรายละเอียดดังนี้
- ถ้าสภาพการจราจร (status) 'คล่องตัว' ค่าเดินทางจะเท่ากับ ระยะทาง (distance) x 10

8

- ถ้าสภาพการจราจร (status) 'ปานกลาง' ค่าเดินทางจะเท่ากับ ระยะทาง (distance) x 12
- ถ้าสภาพการจราจร (status) 'หนาแน่น' ค่าเดินทางจะเท่ากับ ระยะทาง (distance) x 15
- หลังจากนั้นให้พิมพ์ค่าโดยสารออกมา

25. คุณเป็นครูในโรงเรียนแห่งหนึ่ง หลังจากคุณได้เป็นครู คุณได้รู้ว่านอกเหนือจากงานสอนแล้วคุณยังต้องทำงานอื่น ๆ อีกมากมาย โชคดีที่คุณมีความรู้เขียนโปรแกรม คุณจึงอยากเขียนโปรแกรมเพื่อทำงานจิปาถะต่าง ๆ โดยเริ่มต้นจากเขียนโปรแกรมเพื่อประเมินผลการสอบนักเรียน ซึ่งมีรายละเอียดดังนี้

- รับอินพุต 1 ตัวที่เป็นจำนวนเต็ม (score) เพื่อประเมินผลสอบของนักเรียน โดยมีเงื่อนไขดังนี้
- ถ้าคะแนน (score) มากกว่าหรือเท่ากับ 50 ให้พิมพ์ 'Pass'
- นอกจากนั้นให้พิมพ์ 'Fail'

8

26. คุณเป็นอาจารย์ในมหาวิทยาลัยแห่งหนึ่ง หลังจากคุณได้เป็นอาจารย์ คุณได้รู้ว่านอกเหนือจากงานสอนแล้วคุณยังต้องทำงานอื่น ๆ อีกมากมาย โชคดีที่คุณมีความรู้เขียนโปรแกรม คุณจึงอยากเขียนโปรแกรมเพื่อทำงานจิปาถะต่าง ๆ โดยเริ่มต้นจากเขียนโปรแกรมเพื่อตัดเกรดนักศึกษา ซึ่งมีรายละเอียดดังนี้

- รับอินพุต 1 ตัวที่เป็นจำนวนเต็ม (score) เพื่อประเมินผลสอบของนักศึกษา โดยมีเงื่อนไขดังนี้
- ถ้าคะแนน (score) มากกว่าหรือเท่ากับ 80 ให้พิมพ์ 'A'
- ถ้าคะแนน (score) มากกว่าหรือเท่ากับ 70 แต่น้อยกว่า 80 ให้พิมพ์ 'B'
- ถ้าคะแนน (score) มากกว่าหรือเท่ากับ 60 แต่น้อยกว่า 70 ให้พิมพ์ 'C'
- ถ้าคะแนน (score) มากกว่าหรือเท่ากับ 50 แต่น้อยกว่า 60 ให้พิมพ์ 'D'
- นอกจากนั้นให้พิมพ์ 'F'

27. คุณเป็นโปรแกรมเมอร์ในร้านขายของจิปาถะแห่งหนึ่ง ร้านแห่งนี้มีระบบสมาชิกซึ่งผู้เป็นสมาชิกจะได้รับส่วนลดพิเศษ คุณได้รับมอบหมายให้เขียนโปรแกรมซึ่งมีรายละเอียดดังนี้

- รับอินพุต 2 ตัวที่เป็นสายอักขระ และจำนวนจริง (isMember, bill) เพื่อคำนวณส่วนลดจากการเป็นสมาชิกและพิมพ์ออกมา โดยมีเงื่อนไขดังนี้
- ถ้า isMember เท่ากับ 'yes' ให้พิจารณา bill

- ถ้า bill มากกว่าหรือเท่ากับ 500 จะได้รับส่วนลด 5%
- ถ้า bill มากกว่าหรือเท่ากับ 1,000 จะได้รับส่วนลด 10%
- ถ้า bill มากกว่าหรือเท่ากับ 5,000 จะได้รับส่วนลด 15%

• ถ้า isMember เท่ากับ 'no' จะไม่ได้รับส่วนลด

28. คุณเป็นโปรแกรมเมอร์ในธนาคารแห่งหนึ่ง ลูกค้าของธนาคารแห่งนี้ส่วนใหญ่แล้วเป็นคนอเมริกาและคนญี่ปุ่น โดยปกติลูกค้าจะมาใช้บริการแลกเงินบาท คุณได้รับมอบหมายให้เขียนโปรแกรมซึ่งมีรายละเอียดดังนี้

- รับอินพุต 2 ตัวที่เป็นสายอักขระ และจำนวนจริง (currency, amount) เพื่อคำนวณจำนวนเงินบาทที่ลูกค้าแลกได้และพิมพ์ออกมา โดยมีเงื่อนไขดังนี้
- ถ้า currency เท่ากับ 'USD' จำนวนเงินในหน่วยบาทที่แลกได้เป็นดังนี้ (THB=USDx32.5)
- ถ้า currency เท่ากับ 'JPY' จำนวนเงินในหน่วยบาทที่แลกได้เป็นดังนี้ (THB=JPYx0.29)

8

29. คุณเป็นเจ้าของร้านอาหารชื่อพ่อครัวหัวไข่ จุดเด่นของร้านคุณคือเมนูต่าง ๆ ที่ทำจากไข่ ธุรกิจของคุณเพิ่งเริ่มต้นได้ไม่นานและคุณยังไม่มีพนักงานรับออเดอร์ ส่วนตัวคุณเองก็ไม่สามารถรับออเดอร์ได้เพราะคุณวุ่นกับการเป็นพ่อครัว โชคดีที่คุณมีความสามารถในการเขียนโปรแกรม คุณจึงอยากเขียนโปรแกรมเพื่อรับออเดอร์จากลูกค้า ซึ่งมีรายละเอียดดังนี้

- รับอินพุต 2 ตัวที่เป็นสายอักขระ และจำนวนเต็ม (order, amount) เพื่อแจ้งยอดเงินที่ลูกค้าต้องชำระในการสั่งอาหาร โดยเงื่อนไขของโปรแกรมมีดังนี้
- ถ้าลูกค้าสั่ง (order) ไข่ดาว ราคาที่ลูกค้าต้องจ่ายเท่ากับ จำนวนไข่ดาว (amount) x 7
- ถ้าลูกค้าสั่ง (order) ไข่เจียว ราคาที่ลูกค้าต้องจ่ายเท่ากับ จำนวนไข่เจียว (amount) x 10
- ถ้าลูกค้าสั่ง (order) ไข่ต้ม ราคาที่ลูกค้าต้องจ่ายเท่ากับ จำนวนไข่ต้ม (amount) x 5
- นอกจากนั้นให้พิมพ์ 'ร้านเรามี ไข่ดาว, ไข่เจียว, ไข่ต้ม'

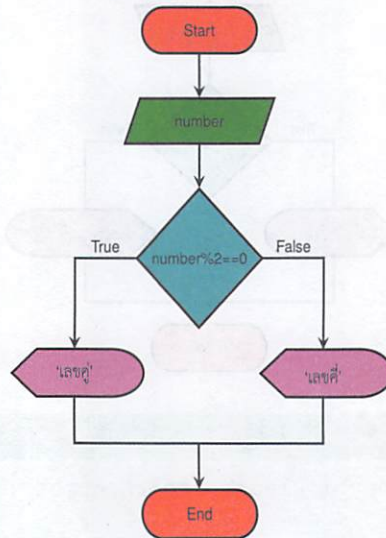
30. คุณเป็นเจ้าของร้านอาหารชื่อพ่อครัวหัวไข่ เนื่องจากคุณเป็นนักธุรกิจที่เก่ง คุณจึงวางแผนเพื่อกระตุ้นยอดขายโดยที่คุณไม่ต้องลงทุนเพิ่ม สิ่งที่คุณคิดได้ก็คือ การแนะนำสินค้าเพิ่มเติมหลังจากลูกค้าสั่งออเดอร์แรกเสร็จ โปรแกรมที่คุณออกแบบมีรายละเอียดดังนี้

- รับอินพุต 1 ตัวที่เป็นสายอักขระ (order) เพื่อแนะนำสินค้าเพิ่มเติม โดยเงื่อนไขของโปรแกรมมีดังนี้
- ถ้าลูกค้าสั่ง (order) ไข่ดาว ให้พิมพ์ 'แนะนำให้สั่งคู่กับต้มจืดไข่น้ำ'
- ถ้าลูกค้าสั่ง (order) ไข่เจียว ให้พิมพ์ 'แนะนำให้สั่งคู่กับไข่ลูกเขย'
- ถ้าลูกค้าสั่ง (order) ไข่ต้ม ให้พิมพ์ 'แนะนำให้สั่งคู่กับยำไข่ดาว'
- นอกจากนั้นให้พิมพ์ 'ร้านเรามี ไข่ดาว, ไข่เจียว, ไข่ต้ม'

8

8.2 Solution

เฉลยข้อที่ 1.



Input:

```
1 number = int(input('Please insert number: '))
2 if number%2 == 0:
3     print('เลขคู่')
4 else:
5     print('เลขคี่')
```

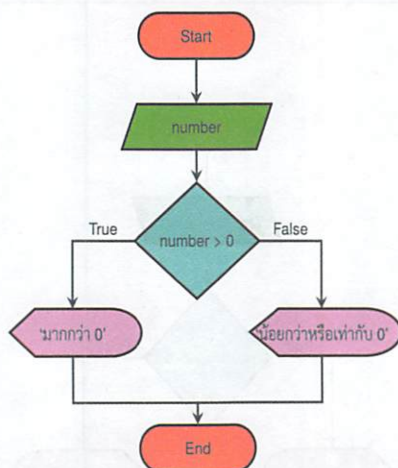
Output:

```
Please insert number: 6
เลขคู่
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนเต็มและกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 2-3 : ถ้า number หารด้วย 2 ลงตัว ให้พิมพ์ 'เลขคู่'
บรรทัดที่ 4-5 : ถ้า number หารด้วย 2 ไม่ลงตัว ให้พิมพ์ 'เลขคี่'

เฉลยข้อที่ 2.



8

Input :

```
1 number = int(input('Please insert number: '))
2 if number > 0:
3     print('มากกว่า 0')
4 else:
5     print('น้อยกว่าหรือเท่ากับ 0')
```

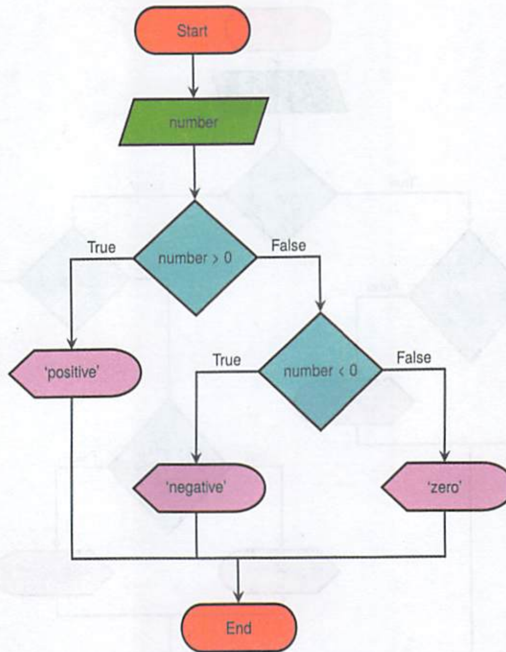
Output :

```
Please insert number : 2
มากกว่า 0
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนเต็มและกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 2-3 : ถ้า number มากกว่า 0 ให้พิมพ์ 'มากกว่า 0'
บรรทัดที่ 4-5 : ถ้า number น้อยกว่าหรือเท่ากับ 0 ให้พิมพ์ 'น้อยกว่าหรือเท่ากับ 0'

เฉลยข้อที่ 3



Input :

```
1 number = float(input('Please insert number: '))
2 if number > 0:
3     print('positive')
4 elif number < 0:
5     print('negative')
6 else:
7     print('zero')
```

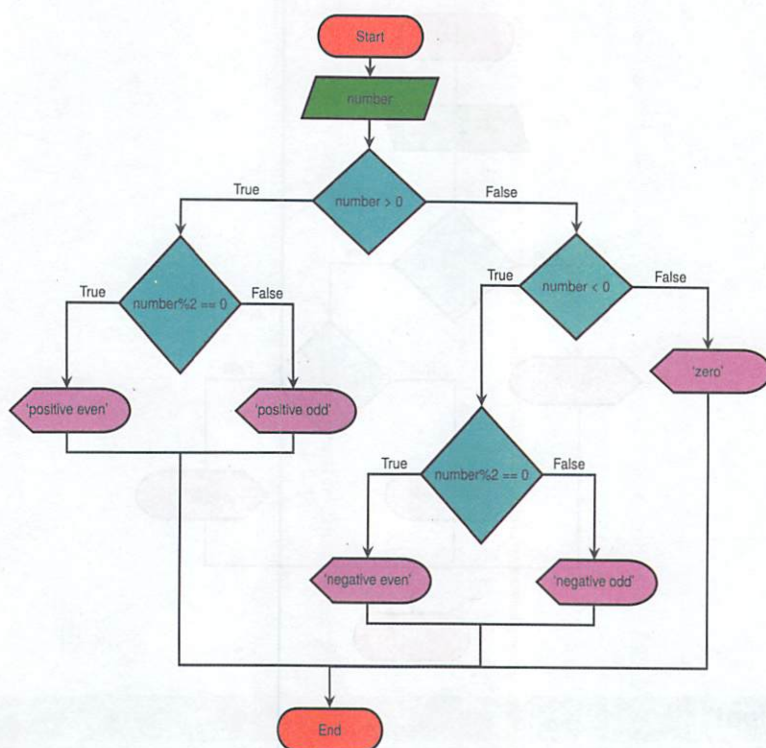
Output :

```
Please insert number: 0
zero
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 2-3 : ถ้า number มากกว่า 0 ให้พิมพ์ 'positive'
บรรทัดที่ 4-5 : ถ้า number น้อยกว่า 0 ให้พิมพ์ 'negative'
บรรทัดที่ 6-7 : ถ้า number เท่ากับ 0 ให้พิมพ์ 'zero'

เฉลยข้อที่ 4.



8

Input :

```
number = float(input('Please insert number: '))
if number > 0:
    if number%2 == 0:
        print('positive even')
    else:
        print('positive odd')
elif number < 0:
    if number%2 == 0:
        print('negative even')
    else:
        print('negative odd')
else:
    print('zero')
```

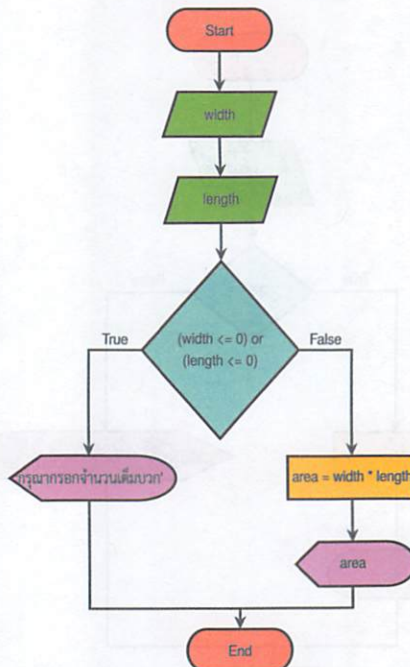
Output :

```
Please insert number: 7
positive odd
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 2 : ถ้า number มากกว่า 0 ให้ทำบรรทัดที่ 3-6
บรรทัดที่ 3-4 : ถ้า number หารด้วย 2 ลงตัว ให้พิมพ์ 'positive even'
บรรทัดที่ 5-6 : ถ้า number หารด้วย 2 ไม่ลงตัว ให้พิมพ์ 'positive odd'
บรรทัดที่ 7 : ถ้า number น้อยกว่า 0 ให้ทำบรรทัดที่ 8-11
บรรทัดที่ 3-4 : ถ้า number หารด้วย 2 ลงตัว ให้พิมพ์ 'negative even'
บรรทัดที่ 5-6 : ถ้า number หารด้วย 2 ไม่ลงตัว ให้พิมพ์ 'negative odd'
บรรทัดที่ 12 : ถ้า number เท่ากับ 0 ให้พิมพ์ 'zero'

เฉลยข้อที่ 5.



Input :

```
width = int(input('Please insert width: '))
length = int(input('Please insert length: '))
if (width <= 0) or (length <= 0):
    print('กรุณากรอกจำนวนเต็มบวก')
else:
    area = width*length
    print(area)
```

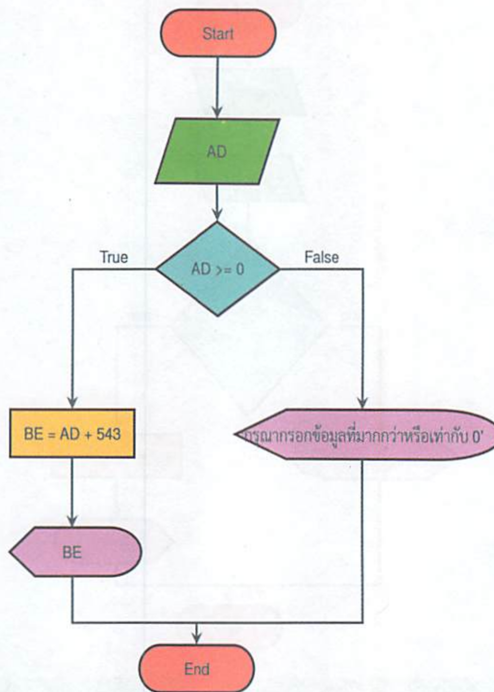
Output :

```
Please insert width: 7
Please insert length: 6
42
```

เฉลยละเอียด

บรรทัดที่ 1-2 : รับอินพุตที่เป็นจำนวนเต็มสองตัวและกำหนดค่าใส่ตัวแปรชื่อ width และ length ตามลำดับ
บรรทัดที่ 3-4 : ถ้า width หรือ length น้อยกว่าหรือเท่ากับ 0 ให้พิมพ์ 'กรุณากรอกจำนวนเต็มบวก'
บรรทัดที่ 5 : ถ้าไม่ใช่ ให้ทำบรรทัดที่ 6-7
บรรทัดที่ 6 : สร้างตัวแปรชื่อ area และกำหนดค่าเท่ากับ width*length
บรรทัดที่ 7 : พิมพ์ค่า area ออกมา

เฉลยข้อที่ 6.



Input :

```
AD = int(input('Please insert AD: '))
if AD >= 0:
    BE = AD + 543
    print(BE)
else:
    print('กรุณากรอกข้อมูลที่มากกว่าหรือเท่ากับ 0')
```

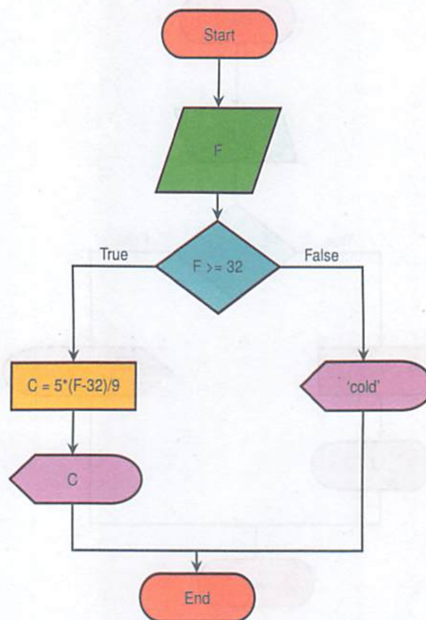
Output :

```
Please insert AD: 0
543
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนเต็มและกำหนดค่าใส่ตัวแปรชื่อ AD
บรรทัดที่ 2 : ถ้า AD มากกว่าเท่ากับ 0 ให้ทำบรรทัดที่ 3-4
บรรทัดที่ 3 : สร้างตัวแปรชื่อ BE และกำหนดค่าเท่ากับ AD + 543
บรรทัดที่ 4 : พิมพ์ค่า BE ออกมา
บรรทัดที่ 5-6 : ถ้า AD น้อยกว่า 0 ให้พิมพ์ 'กรุณากรอกข้อมูลที่มากกว่าหรือเท่ากับ 0'

เฉลยข้อที่ 7.



Input :

```
F = float(input('Please insert F: '))
if F >= 32:
    C = 5*(F-32)/9
    print(C)
else:
    print('cold')
```

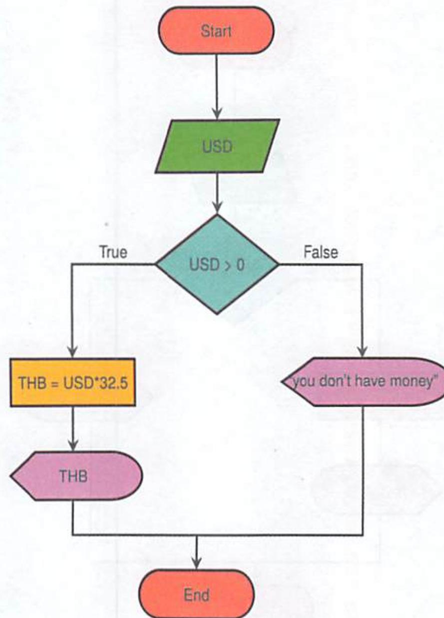
Output :

```
Please insert F: 64
17.77777777777778
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ F
บรรทัดที่ 2 : ถ้า F มากกว่าหรือเท่ากับ 32 ให้ทำบรรทัดที่ 3-4
บรรทัดที่ 3 : สร้างตัวแปรชื่อ C และกำหนดค่าเท่ากับ 5*(F-32)/9
บรรทัดที่ 4 : พิมพ์ค่า C ออกมา
บรรทัดที่ 5-6 : ถ้า F น้อยกว่า 32 ให้พิมพ์ 'cold'

เฉลยข้อที่ 8.



Input :

```
USD = float(input('Please insert USD: '))
if USD > 0:
    THB = USD*32.5
    print(THB)
else:
    print("you don't have money")
```

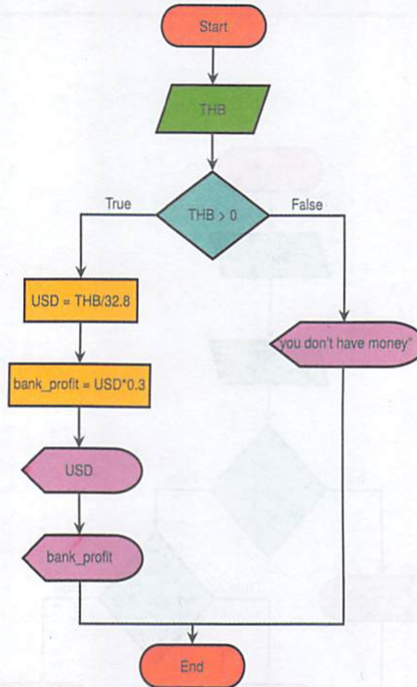
Output :

```
Please insert USD: -7
you don't have money
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ USD
บรรทัดที่ 2 : ถ้า USD มากกว่า 0 ให้ทำบรรทัดที่ 3-4
บรรทัดที่ 3 : สร้างตัวแปรชื่อ THB และกำหนดค่าเท่ากับ USD*32.5
บรรทัดที่ 4 : พิมพ์ค่า THB ออกมา
บรรทัดที่ 5-6 : ถ้า USD น้อยกว่าหรือเท่ากับ 0 ให้พิมพ์ "you don't have money"

เฉลยข้อที่ 9.



Input :

```
THB = float(input('Please insert THB: '))
if THB > 0:
    USD = THB/32.8
    bank_profit = USD*0.3
    print('USD:',USD)
    print('profit:', bank_profit)
else:
    print("you don't have money")
```

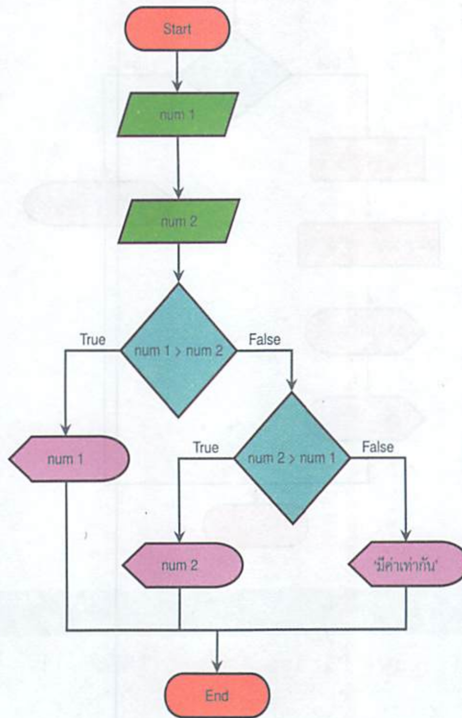
Output :

```
Please insert THB: 84375
USD: 2572.408536585366
profit: 771.7225609756098
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ THB
บรรทัดที่ 2 : ถ้า THB มากกว่า 0 ให้ทำบรรทัดที่ 3-6
บรรทัดที่ 3 : สร้างตัวแปรชื่อ USD และกำหนดค่าเท่ากับ THB/32.8
บรรทัดที่ 4 : สร้างตัวแปรชื่อ bank_profit และกำหนดค่าเท่ากับ USD*0.3
บรรทัดที่ 5 : พิมพ์ค่า USD ออกมา
บรรทัดที่ 6 : พิมพ์ค่า bank_profit ออกมา
บรรทัดที่ 7-8 : ถ้า THB น้อยกว่าหรือเท่ากับ 0 ให้พิมพ์ “you don’t have money”

เฉลยข้อที่ 10.



Input :

```
num1 = float(input('Please insert num1: '))
num2 = float(input('Please insert num2: '))
if num1 > num2:
    print(num1)
elif num2 > num1:
    print(num2)
else:
    print('มีค่าเท่ากัน')
```

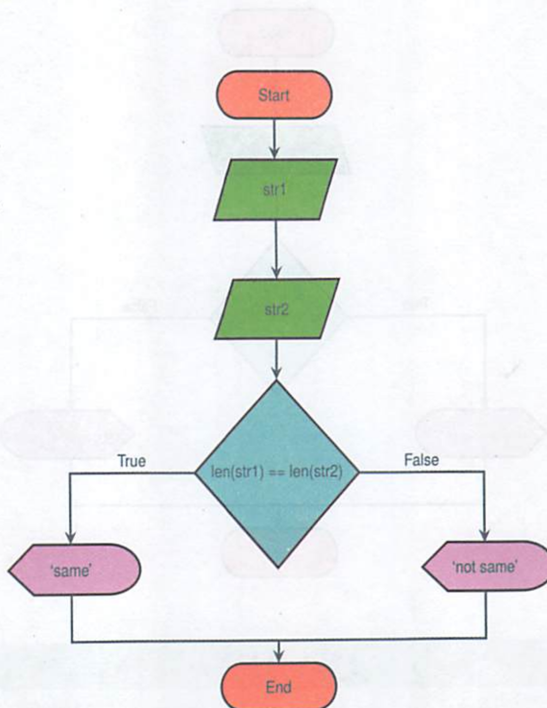
Output :

```
Please insert num1: 7
Please insert num2: 6
7.0
```

เฉลยละเอียด

บรรทัดที่ 1-2 : รับอินพุตที่เป็นจำนวนจริงสองตัวและกำหนดค่าใส่ตัวแปรชื่อ num1 และ num2
บรรทัดที่ 3-4 : ถ้า num1 มากกว่า num2 ให้พิมพ์ค่า num1 ออกมา
บรรทัดที่ 5-6 : ถ้า num2 มากกว่า num1 ให้พิมพ์ค่า num2 ออกมา
บรรทัดที่ 7-8 : ถ้า num1 เท่ากับ num2 ให้พิมพ์ 'มีค่าเท่ากัน'

เฉลยข้อที่ 11.



Input :

```
str1 = input('Please insert str1: ')
str2 = input('Please insert str2: ')
if len(str1) == len(str2):
    print('same')
else:
    print('not same')
```

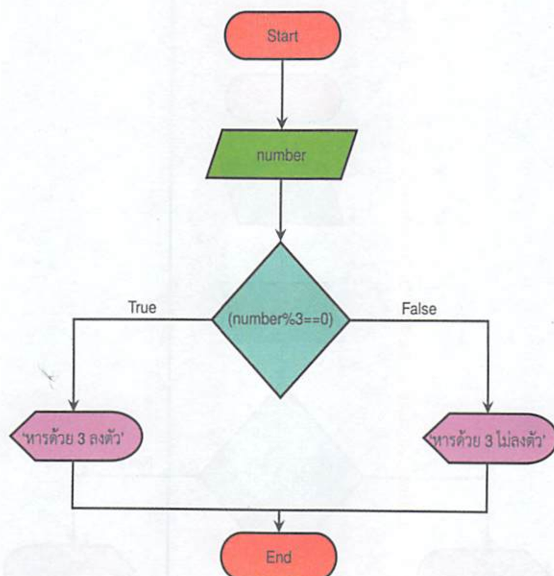
Output :

```
Please insert str1: abc
Please insert str2: cbd
same
```

เฉลยละเอียด

บรรทัดที่ 1-2 : รับอินพุตที่เป็น string สองตัวและกำหนดค่าใส่ตัวแปรชื่อ str1 และ str2
บรรทัดที่ 3-4 : ถ้าความยาว str1 และ str2 เท่ากัน ให้พิมพ์ 'same'
บรรทัดที่ 5-6 : ถ้าความยาว str1 และ str2 ไม่เท่ากัน ให้พิมพ์ 'not same'

เฉลยข้อที่ 12.



Input :

```
number = int(input('Please insert number: '))
if number%3 == 0:
    print('หารด้วย 3 ลงตัว')
else:
    print('หารด้วย 3 ไม่ลงตัว')
```

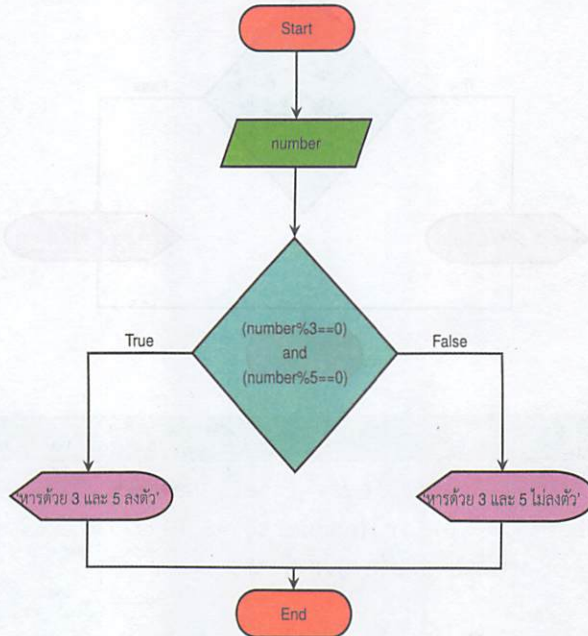
Output :

```
Please insert number: 20
หารด้วย 3 ไม่ลงตัว
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนเต็มและกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 2-3 : ถ้า number หารด้วย 3 ลงตัว ให้พิมพ์ 'หารด้วย 3 ลงตัว'
บรรทัดที่ 4-5 : ถ้า number หารด้วย 3 ไม่ลงตัว ให้พิมพ์ 'หารด้วย 3 ไม่ลงตัว'

เฉลยข้อที่ 13.



8

Input :

```
1 number = int(input('Please insert number: '))
2 if (number%3 == 0) and (number%5 == 0):
3     print('หารด้วย 3 และ 5 ลงตัว')
4 else:
5     print('หารด้วย 3 และ 5 ไม่ลงตัว')
```

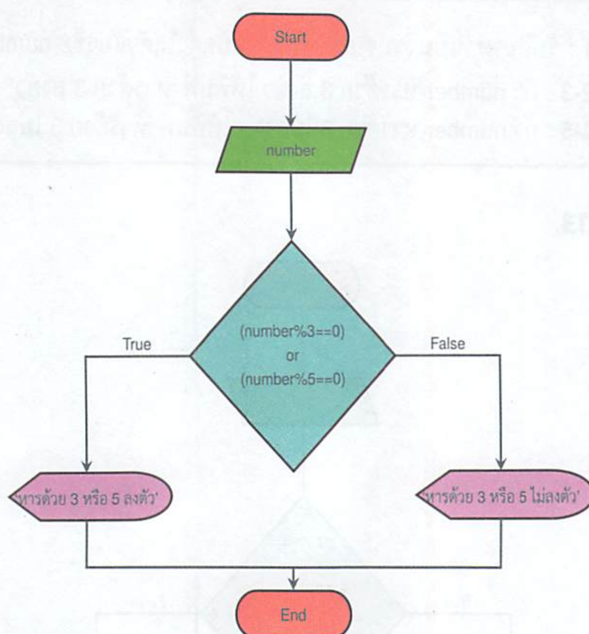
Output :

```
Please insert number: 14
หารด้วย 3 และ 5 ไม่ลงตัว
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนเต็มและกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 2-3 : ถ้า number หารด้วย 3 และ 5 ลงตัว ให้พิมพ์ 'หารด้วย 3 และ 5 ลงตัว'
บรรทัดที่ 4-5 : ถ้า number หารด้วย 3 และ 5 ไม่ลงตัว ให้พิมพ์ 'หารด้วย 3 และ 5 ไม่ลงตัว'

เฉลยข้อที่ 14.



Input :

```
1 number = int(input('Please insert number: '))
2 if (number%3 == 0) or (number%5 == 0):
3     print('หารด้วย 3 หรือ 5 ลงตัว')
4 else:
5     print('หารด้วย 3 หรือ 5 ไม่ลงตัว')
```

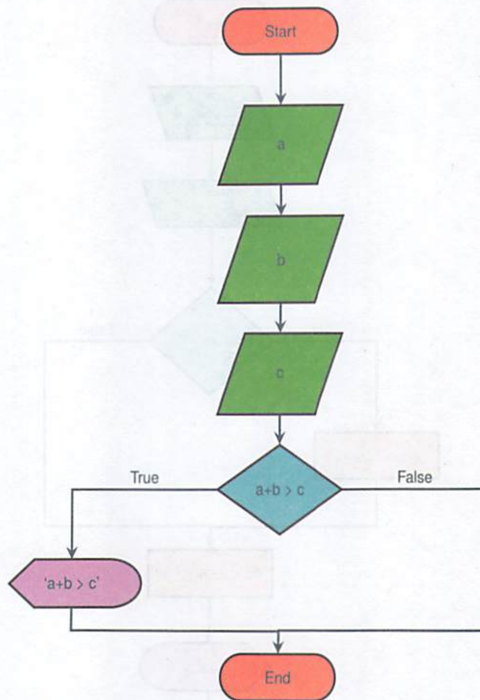
Output :

```
Please insert number: 11
หารด้วย 3 หรือ 5 ไม่ลงตัว
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนเต็มและกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 2-3 : ถ้า number หารด้วย 3 หรือ 5 ลงตัว ให้พิมพ์ 'หารด้วย 3 หรือ 5 ลงตัว'
บรรทัดที่ 4-5 : ถ้า number หารด้วย 3 หรือ 5 ไม่ลงตัว ให้พิมพ์ 'หารด้วย 3 หรือ 5 ไม่ลงตัว'

เฉลยข้อที่ 15.



Input :

```
1 a = float(input('Please insert a: '))
2 b = float(input('Please insert b: '))
3 c = float(input('Please insert c: '))
4 if a + b > c:
5     print('a + b > c')
```

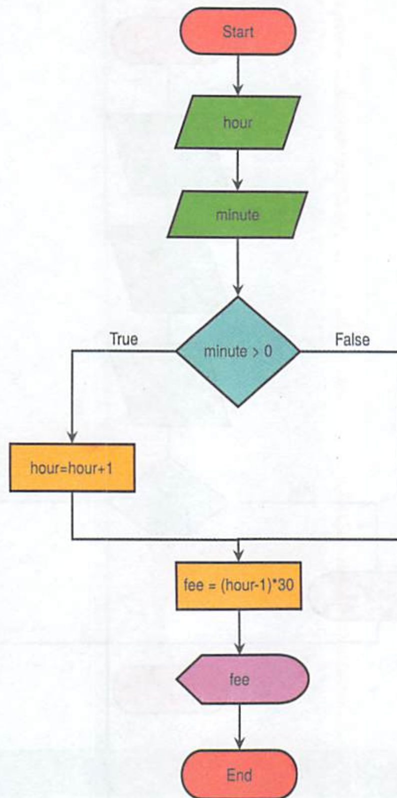
Output :

```
Please insert a: 1
Please insert b: 2
Please insert c: 3
```

เฉลยละเอียด

บรรทัดที่ 1-3 : รับอินพุตที่เป็นจำนวนจริงสามตัวและกำหนดค่าใส่ตัวแปรชื่อ a, b และ c
บรรทัดที่ 4-5 : ถ้า a+b มากกว่า c ให้พิมพ์ 'a + b > c'

เฉลยข้อที่ 16.



Input :

```
1 hour = int(input('Please insert hour: '))
2 minute = int(input('Please insert minute: '))
3 if minute > 0:
4     hour = hour + 1
5 fee = (hour - 1)*30
6 print(fee)
```

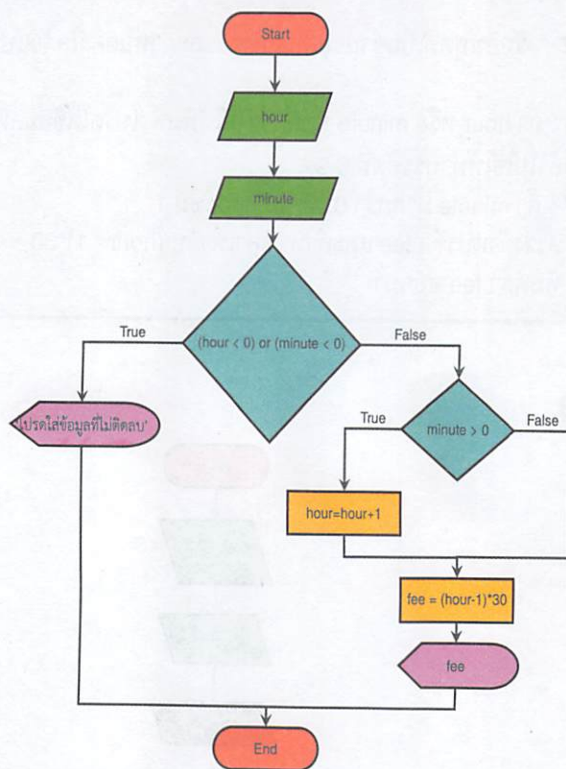
Output :

```
Please insert hour: 1
Please insert minute: 2
30
```

เฉลยละเอียด

บรรทัดที่ 1-2 : รับอินพุตที่เป็นจำนวนเต็มสองตัวและกำหนดค่าใส่ตัวแปรชื่อ hour และ minute
บรรทัดที่ 3-4 : ถ้า minute มากกว่า 0 ให้เพิ่ม hour ขึ้น 1
บรรทัดที่ 5 : สร้างตัวแปรชื่อ fee และกำหนดค่าเท่ากับ (hour - 1)*30
บรรทัดที่ 6 : พิมพ์ค่า fee ออกมา

เฉลยข้อที่ 17.



Input :

```
hour = int(input('Please insert hour: '))
minute = int(input('Please insert minute: '))
if (hour < 0) or (minute < 0):
    print('โปรดใส่ข้อมูลที่ไม่ติดลบ')
else:
    if minute > 0:
        hour = hour + 1
    fee = (hour - 1)*30
    print(fee)
```

Output :

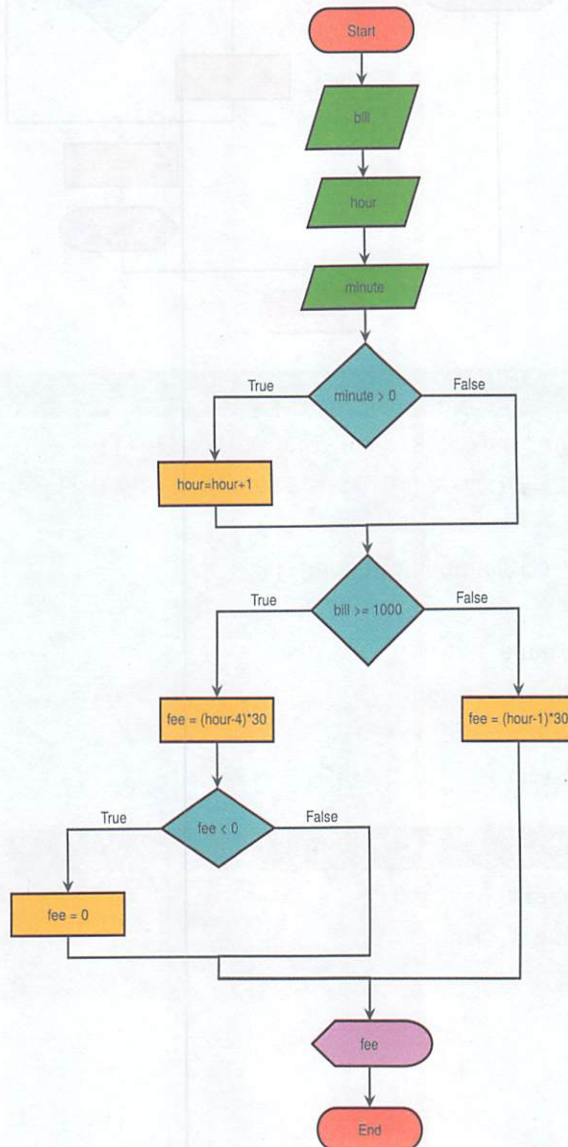
```
Please insert hour: 7
Please insert minute: 6
210
```

8

เฉลยละเอียด

บรรทัดที่ 1-2 : รับอินพุตที่เป็นจำนวนเต็มสองตัวและกำหนดค่าใส่ตัวแปรชื่อ hour และ minute
บรรทัดที่ 3-4 : ถ้า hour หรือ minute น้อยกว่า 0 ให้พิมพ์ 'โปรดใส่ข้อมูลที่ไม่ติดลบ'
บรรทัดที่ 5 : ถ้าไม่ใช่ให้ทำบรรทัดที่ 6-9
บรรทัดที่ 6-7 : ถ้า minute มากกว่า 0 ให้เพิ่ม hour ขึ้น 1
บรรทัดที่ 8 : สร้างตัวแปรชื่อ fee และกำหนดค่าเท่ากับ (hour - 1)*30
บรรทัดที่ 9 : พิมพ์ค่า fee ออกมา

เฉลยข้อที่ 18.



8

Input :

```
1 bill = int(input('Please insert bill: '))
2 hour = int(input('Please insert hour: '))
3 minute = int(input('Please insert minute: '))
4 if minute > 0:
5     hour = hour + 1
6 if bill >= 1000:
7     fee = (hour - 4)*30
8     if fee < 0:
9         fee = 0
10 else:
11     fee = (hour - 1)*30
12 print(fee)
```

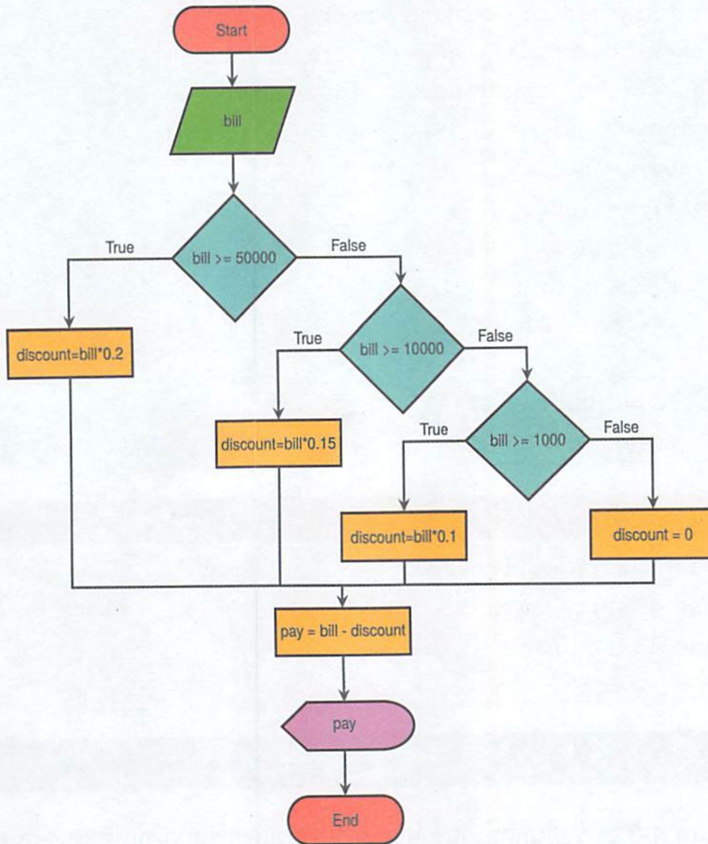
Output :

```
Please insert bill: 5000
Please insert hour: 5
Please insert minute: 5
60
```


เฉลยละเอียด

บรรทัดที่ 1-3 : รับอินพุตที่เป็นจำนวนเต็มสามตัวและกำหนดค่าใส่ตัวแปรชื่อ bill, hour และ minute
บรรทัดที่ 4-5 : ถ้า minute มากกว่า 0 ให้เพิ่ม hour ขึ้น 1
บรรทัดที่ 6 : ถ้า bill มากกว่าหรือเท่ากับ 1000 ให้ทำบรรทัดที่ 7-9
บรรทัดที่ 7 : สร้างตัวแปรชื่อ fee เพื่อคำนวณค่าจอดรถโดยหักลบ hour ไป 4 ชั่วโมง
บรรทัดที่ 8-9 : ถ้า fee น้อยกว่า 0 กำหนดให้ fee เท่ากับ 0
บรรทัดที่ 10 : ถ้า bill น้อยกว่า 1000 ให้ทำบรรทัดที่ 11
บรรทัดที่ 11 : สร้างตัวแปรชื่อ fee เพื่อคำนวณค่าจอดรถโดยหักลบ hour ไป 1 ชั่วโมง
บรรทัดที่ 12 : พิมพ์ค่า fee ออกมา

เฉลยข้อที่ 19.



8

Input :

```
bill = float(input('Please insert bill: '))
if bill >= 50000:
    discount = bill*0.2
elif bill >= 10000:
    discount = bill*0.15
elif bill >= 1000:
    discount = bill*0.1
else:
    discount = 0
pay = bill - discount
print(pay)
```

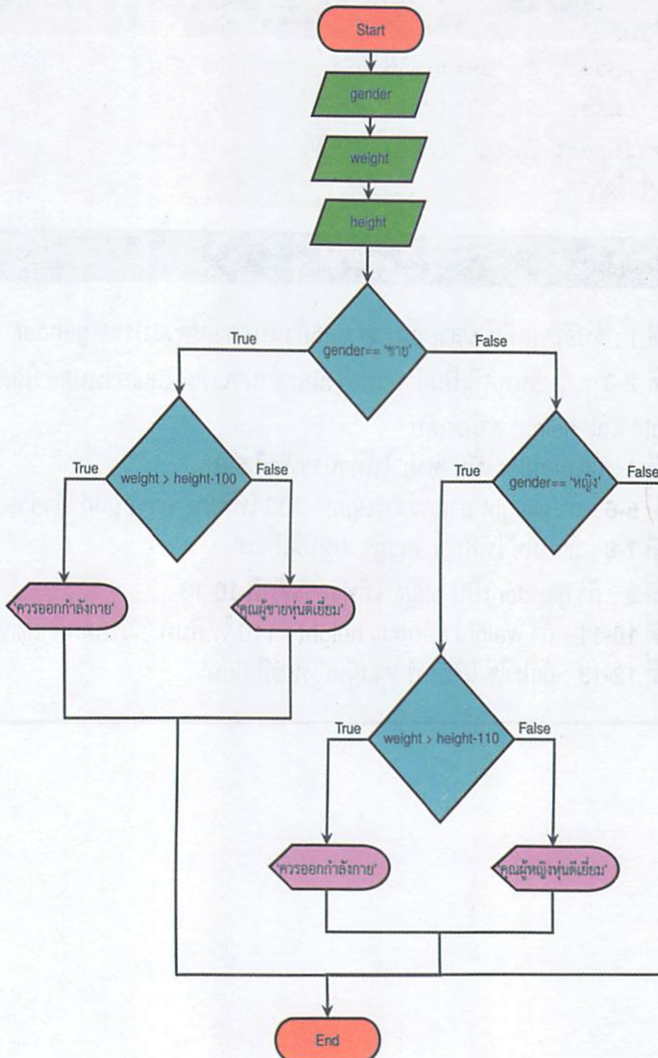
Output :

```
Please insert bill: 200000
160000.0
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ bill
บรรทัดที่ 2-3 : ถ้า bill มากกว่าเท่ากับ 50000 ให้คำนวณราคาสินค้าที่ลดไป 20% แล้วเก็บไว้ในตัวแปร discount
บรรทัดที่ 4-5 : ถ้า bill มีค่ามากกว่า 10000 แต่ไม่ถึง 50000 ให้คำนวณราคาสินค้าที่ลดไป 15% แล้วเก็บไว้ในตัวแปร discount
บรรทัดที่ 6-7 : ถ้า bill มีค่ามากกว่า 1000 แต่ไม่ถึง 10000 ให้คำนวณราคาสินค้าที่ลดไป 10% แล้วเก็บไว้ในตัวแปร discount
บรรทัดที่ 8-9 : ถ้า bill น้อยกว่า 1000 ให้ discount เป็น 0
บรรทัดที่ 10 : สร้างตัวแปรชื่อ pay และกำหนดค่าเท่ากับ bill - discount
บรรทัดที่ 11 : พิมพ์ค่า pay ออกมา

เฉลยข้อที่ 20.



8

Input :

```
1 gender = input('Please insert gender: ')
2 weight = float(input('Please insert weight(kg): '))
3 height = int(input('Please insert height(cm): '))
4 if gender == 'ชาย':
5     if weight > height-100:
6         print('ควรออกกำลังกาย')
7     else:
8         print('คุณผู้ชายหุ่นดีเยี่ยม')
9 elif gender == 'หญิง':
10     if weight > height-110:
11         print('ควรออกกำลังกาย')
12     else:
13         print('คุณผู้หญิงหุ่นดีเยี่ยม')
```

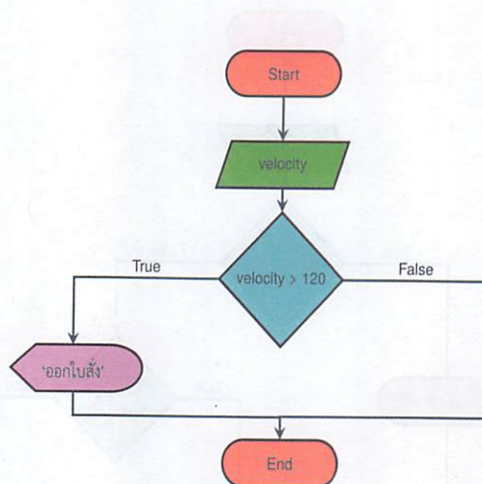
Output :

```
Please insert gender: ชาย
Please insert weight(kg): 95
Please insert height(cm): 180
ควรออกกำลังกาย
```


เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ gender
บรรทัดที่ 2-3 : รับอินพุตที่เป็นจำนวนจริงและจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ weight และ height ตามลำดับ
บรรทัดที่ 4 : ถ้า gender เป็น 'ชาย' ให้ทำบรรทัดที่ 5-8
บรรทัดที่ 5-6 : ถ้า weight มากกว่า height - 100 ให้พิมพ์ 'ควรออกกำลังกาย'
บรรทัดที่ 7-8 : ถ้าไม่ใช่ ให้พิมพ์ 'คุณผู้ชายหุ่นดีเยี่ยม'
บรรทัดที่ 9 : ถ้า gender เป็น 'หญิง' ให้ทำบรรทัดที่ 10-13
บรรทัดที่ 10-11 : ถ้า weight มากกว่า height - 110 ให้พิมพ์ 'ควรออกกำลังกาย'
บรรทัดที่ 12-13 : ถ้าไม่ใช่ ให้พิมพ์ 'คุณผู้หญิงหุ่นดีเยี่ยม'

เฉลยข้อที่ 21.



Input :

```
1 velocity = float(input('Please insert velocity(km/hr): ')
    ↪ )
2 if velocity > 120:
3     print('ออกใบสั่ง')
```

Output :

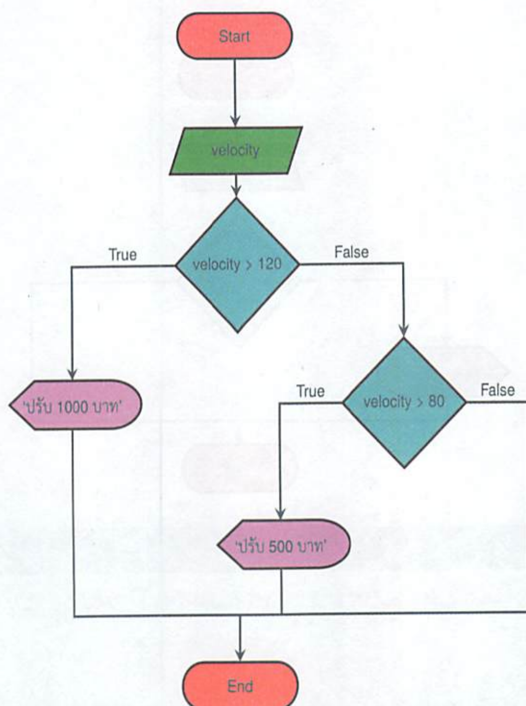
```
Please insert velocity(km/hr): 135
ออกใบสั่ง
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ velocity

บรรทัดที่ 2-3 : ถ้า velocity มากกว่า 120 ให้พิมพ์ 'ออกใบสั่ง'

เฉลยข้อที่ 22.



Input:

```
1 velocity = float(input('Please insert velocity(km/hr): ')
    ↪ )
2 if velocity > 120:
3     print('ปรับ 1,000 บาท')
4 elif velocity > 80:
5     print('ปรับ 500 บาท')
```

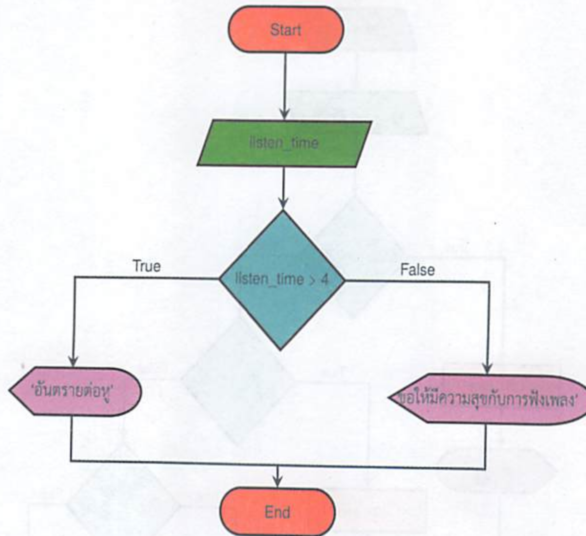
Output:

```
Please insert velocity(km/hr): 600
ปรับ 1,000 บาท
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ velocity
บรรทัดที่ 2-3 : ถ้า velocity มากกว่า 120 ให้พิมพ์ 'ปรับ 1,000 บาท'
บรรทัดที่ 3-4 : ถ้า velocity มากกว่า 80 แต่ไม่ถึง 120 ให้พิมพ์ 'ปรับ 500 บาท'

เฉลยข้อที่ 23.



Input :

```
1 listen_time = float(input('Please insert listen_time(hour): '))
2 if listen_time > 4:
3     print('อันตรายต่อหู')
4 else:
5     print('ขอให้มีความสุขกับการฟังเพลง')
```

Output :

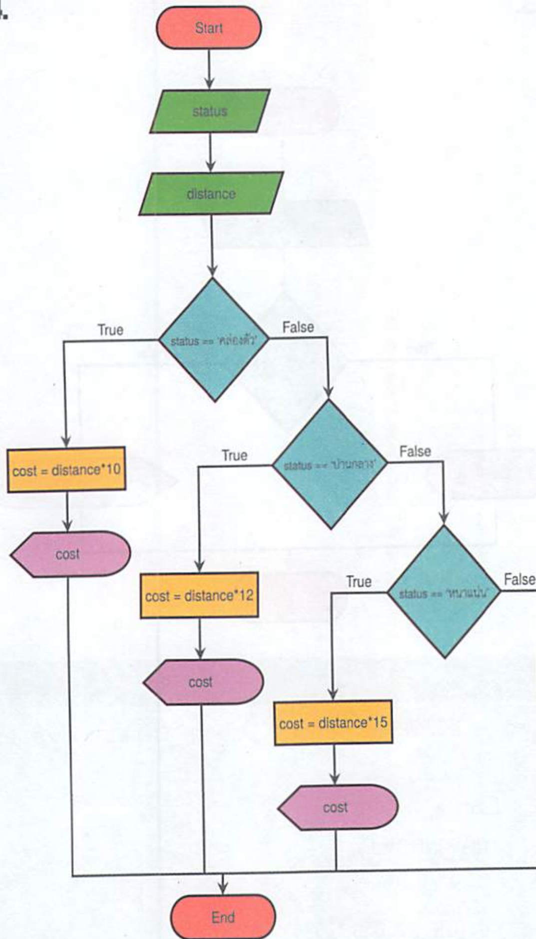
```
Please insert listen_time(hour): 4.1
อันตรายต่อหู
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ listen_time
บรรทัดที่ 2-3 : ถ้า listen_time มากกว่า 4 ให้พิมพ์ 'อันตรายต่อหู'
บรรทัดที่ 4-5 : ถ้าไม่ใช่ ให้พิมพ์ 'ขอให้มีความสุขกับการฟังเพลง'

8

เฉลยข้อที่ 24.



Input :

```python
traffic_status = input('Please insert traffic_status: ')
distance = float(input('Please insert distance(km): '))
if traffic_status == 'คล่องตัว':
    cost = distance*10
    print(cost)
elif traffic_status == 'ปานกลาง':
    cost = distance*12
    print(cost)
elif traffic_status == 'หนาแน่น':
    cost = distance*15
    print(cost)
```

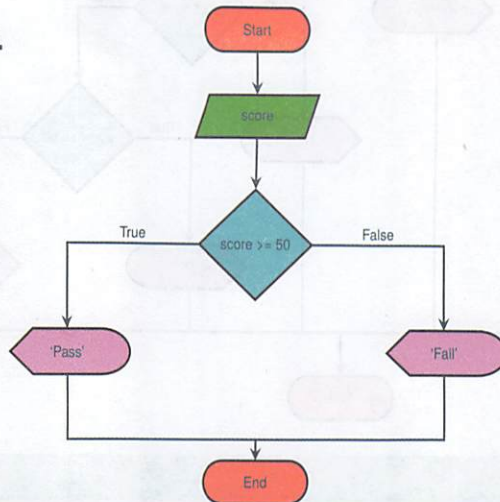
Output :

```
Please insert status: คล่องตัว
Please insert distance(km): 20
200.0
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ traffic_status
บรรทัดที่ 2 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ distance
บรรทัดที่ 3 : ถ้า traffic_status เป็น 'คล่องตัว' ให้ทำบรรทัดที่ 4-5
บรรทัดที่ 4-5 : สร้างตัวแปรชื่อ cost เพื่อคำนวณค่าเดินทางตามที่โจทย์กำหนด พิมพ์ค่า cost ออกมา
บรรทัดที่ 6 : ถ้า traffic_status เป็น 'ปานกลาง' ให้ทำบรรทัดที่ 4-5
บรรทัดที่ 7-8 : สร้างตัวแปรชื่อ cost เพื่อคำนวณค่าเดินทางตามที่โจทย์กำหนด พิมพ์ค่า cost ออกมา
บรรทัดที่ 9 : ถ้า traffic_status เป็น 'หนาแน่น' ให้ทำบรรทัดที่ 4-5
บรรทัดที่ 10-11 : สร้างตัวแปรชื่อ cost เพื่อคำนวณค่าเดินทางตามที่โจทย์กำหนด พิมพ์ค่า cost ออกมา

เฉลยข้อที่ 25.



Input :

```
1 score = int(input('Please insert score: '))
2 if score >= 50:
3     print('Pass')
4 else:
5     print('Fail')
```

Output :

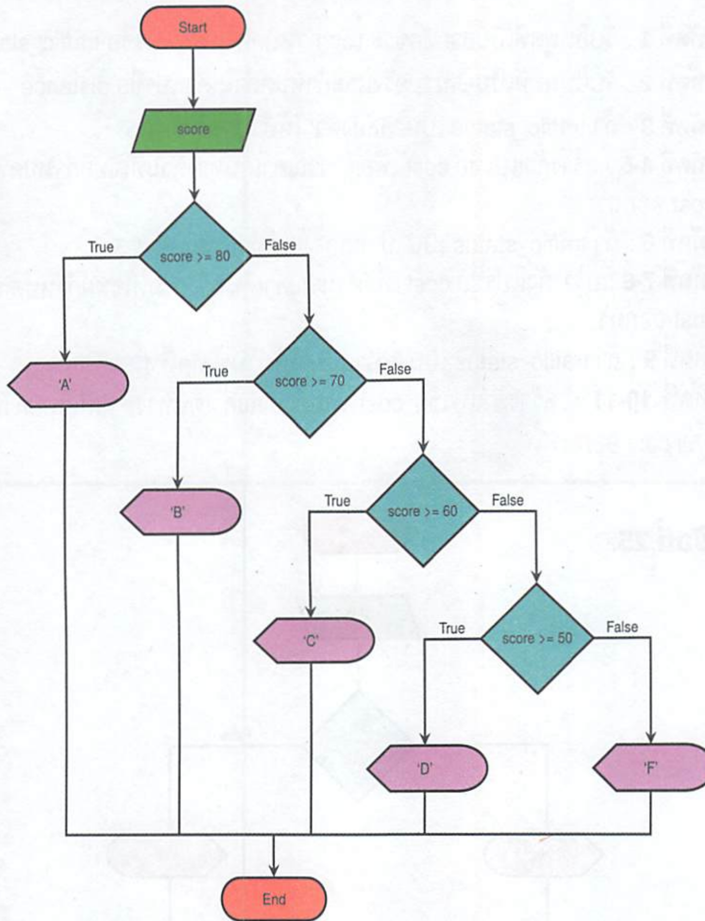
```
Please insert score: 78
Pass
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนเต็มและกำหนดค่าใส่ตัวแปรชื่อ score
บรรทัดที่ 2-3 : ถ้า score มากกว่าหรือเท่ากับ 50 ให้พิมพ์ 'Pass'
บรรทัดที่ 4-5 : ถ้า score น้อยกว่า 50 ให้พิมพ์ 'Fail'

8

เฉลยข้อที่ 26.



Input :

```
score = int(input('Please insert score: '))
if score >= 80:
    print('A')
elif score >= 70:
    print('B')
elif score >= 60:
    print('C')
elif score >= 50:
    print('D')
else:
    print('F')
```

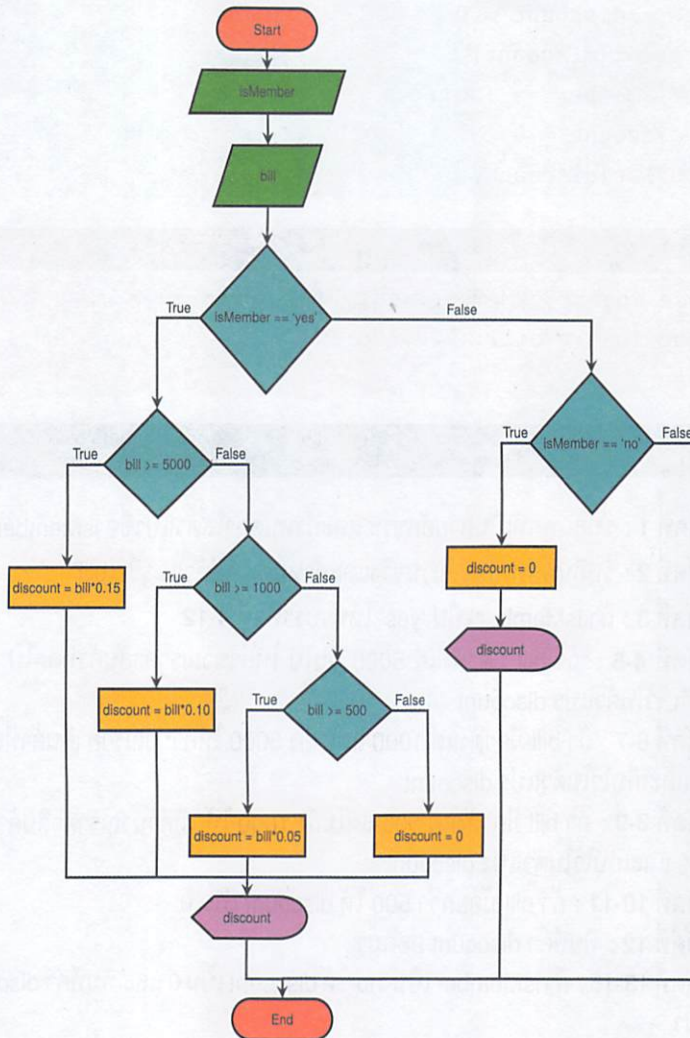
Output :

```
Please insert score: 84
A
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนเต็มและกำหนดค่าใส่ตัวแปรชื่อ score
บรรทัดที่ 2-3 : ถ้า score มีค่าตั้งแต่ 80 ขึ้นไป ให้พิมพ์ 'A'
บรรทัดที่ 4-5 : ถ้า score มีค่าตั้งแต่ 70 ถึง 79 ให้พิมพ์ 'B'
บรรทัดที่ 6-7 : ถ้า score มีค่าตั้งแต่ 60 ถึง 69 ให้พิมพ์ 'C'
บรรทัดที่ 8-9 : ถ้า score มีค่าตั้งแต่ 50 ถึง 59 ให้พิมพ์ 'D'
บรรทัดที่ 10-11 : ถ้า score มีค่าน้อยกว่า 50 ให้พิมพ์ 'F'

เฉลยข้อที่ 27.



8

Input :

```
1  isMember = input('Please insert isMember: ')
2  bill = float(input('Please insert bill: '))
3  if isMember == 'yes':
4      if bill >= 5000:
5          discount = bill*0.15
6      elif bill >= 1000:
7          discount = bill*0.10
8      elif bill >= 500:
9          discount = bill*0.05
10     else:
11         discount = 0
12     print(discount)
13 elif isMember == 'no':
14     discount = 0
15     print(discount)
```

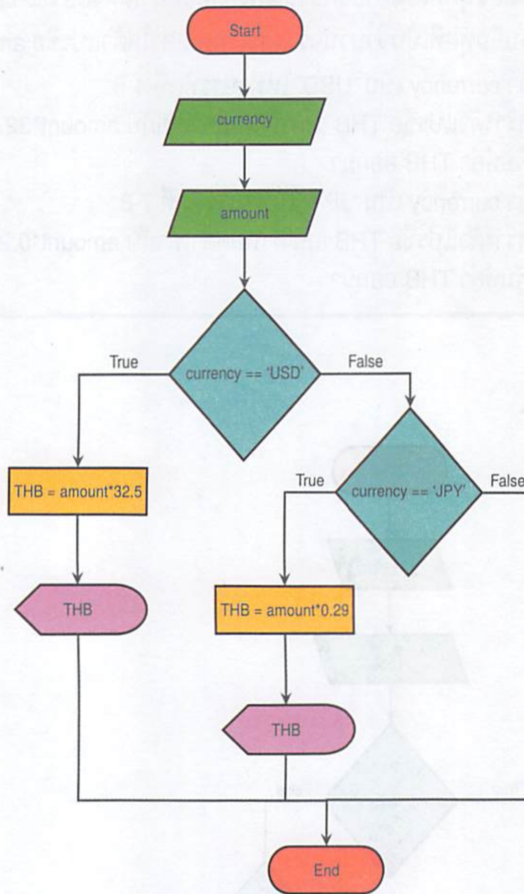
Output :

```
Please insert isMember: no
Please insert bill: 700
0
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ isMember
บรรทัดที่ 2 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ bill
บรรทัดที่ 3 : ถ้า isMember เป็น 'yes' ให้ทำบรรทัดที่ 4-12
บรรทัดที่ 4-5 : ถ้า bill มีค่าตั้งแต่ 5000 ขึ้นไป ให้คำนวณราคาสินค้าที่ลดไป 15% และเก็บไว้ในตัวแปร discount
บรรทัดที่ 6-7 : ถ้า bill มีค่าตั้งแต่ 1000 แต่ไม่ถึง 5000 ให้คำนวณราคาสินค้าที่ลดไป 10% และเก็บไว้ในตัวแปร discount
บรรทัดที่ 8-9 : ถ้า bill มีค่าตั้งแต่ 500 แต่ไม่ถึง 1000 ให้ ให้คำนวณราคาสินค้าที่ลดไป 5% และเก็บไว้ในตัวแปร discount
บรรทัดที่ 10-11 : ถ้า bill น้อยกว่า 500 ให้ discount เป็น 0
บรรทัดที่ 12 : พิมพ์ค่า discount ออกมา
บรรทัดที่ 13-15 : ถ้า isMember เป็น 'no' ให้ discount เป็น 0 และ พิมพ์ค่า discount ออกมา

เฉลยข้อที่ 28.



Input :

```
currency = input('Please insert currency: ')
amount = float(input('Please insert amount: '))
if currency == 'USD':
    THB = amount*32.5
    print(THB)
elif currency == 'JPY':
    THB = amount*0.29
    print(THB)
```

Output :

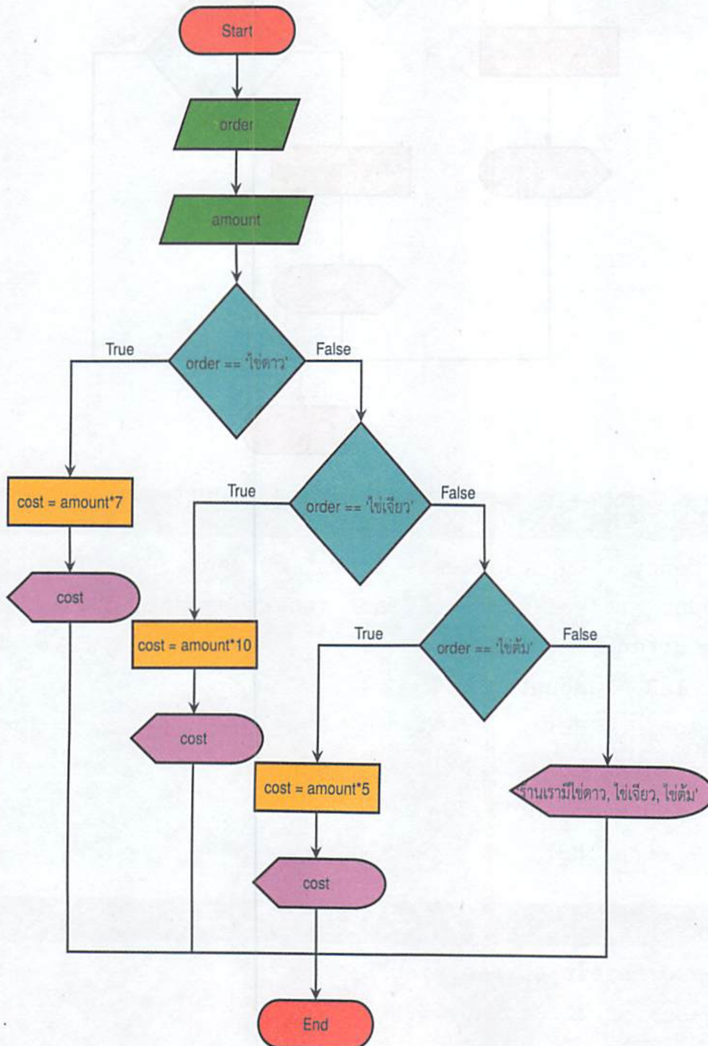
```
Please insert currency: USD
Please insert amount: 100
3250.0
```


8

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ currency
บรรทัดที่ 2 : รับอินพุตที่เป็นจำนวนจริงและกำหนดค่าใส่ตัวแปรชื่อ amount
บรรทัดที่ 3 : ถ้า currency เป็น 'USD' ให้ทำบรรทัดที่ 4-5
บรรทัดที่ 4 : สร้างตัวแปรชื่อ THB และกำหนดค่าเท่ากับ amount*32.5
บรรทัดที่ 5 : พิมพ์ค่า THB ออกมา
บรรทัดที่ 6 : ถ้า currency เป็น 'JPY' ให้ทำบรรทัดที่ 7-8
บรรทัดที่ 7 : สร้างตัวแปรชื่อ THB และกำหนดค่าเท่ากับ amount*0.29
บรรทัดที่ 8 : พิมพ์ค่า THB ออกมา

เฉลยข้อที่ 29.



Input :

```
1  order = input('Please order: ')
2  amount = int(input('Please insert amount: '))
3  if order == 'ไข่ดาว':
4      cost = amount*7
5      print(cost)
6  elif order == 'ไข่เจียว':
7      cost = amount*10
8      print(cost)
9  elif order == 'ไข่ต้ม':
10     cost = amount*5
11     print(cost)
12 else:
13     print('ร้านเรามีไข่ดาว, ไข่เจียว, ไข่ต้ม')
```

Output :

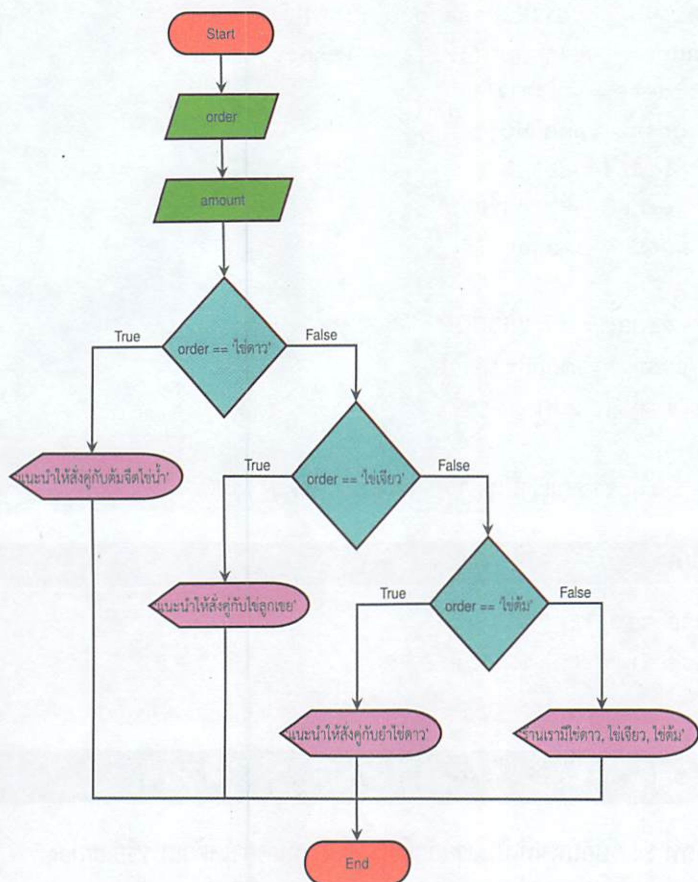
```
Please order: ไข่ดาว
Please insert amount: 2
14
```


เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ order
บรรทัดที่ 2 : รับอินพุตที่เป็นจำนวนเต็มและกำหนดค่าใส่ตัวแปรชื่อ amount
บรรทัดที่ 3 : ถ้า order เป็น 'ไข่ดาว' ให้ทำบรรทัดที่ 4-5
บรรทัดที่ 4 : สร้างตัวแปรชื่อ cost และกำหนดค่าเท่ากับ amount*7
บรรทัดที่ 5 : พิมพ์ค่า cost ออกมา
บรรทัดที่ 6 : ถ้า order เป็น 'ไข่เจียว' ให้ทำบรรทัดที่ 7-8
บรรทัดที่ 7 : สร้างตัวแปรชื่อ cost และกำหนดค่าเท่ากับ amount*10
บรรทัดที่ 8 : พิมพ์ค่า cost ออกมา
บรรทัดที่ 9 : ถ้า order เป็น 'ไข่ต้ม' ให้ทำบรรทัดที่ 10-11
บรรทัดที่ 10 : สร้างตัวแปรชื่อ cost และกำหนดค่าเท่ากับ amount*5
บรรทัดที่ 11 : พิมพ์ค่า cost ออกมา
บรรทัดที่ 12-13 : ถ้า order เป็นอื่น ๆ ให้พิมพ์ 'ร้านเรามีไข่ดาว, ไข่เจียว, ไข่ต้ม'

8

เฉลยข้อที่ 30.



Input :

```
order = input('Please order: ')
if order == 'ไข่ดาว':
    print('แนะนำให้สั่งคู่กับต้มจืดไข่น้ำ')
elif order == 'ไข่เจียว':
    print('แนะนำให้สั่งคู่กับไข่ลูกเขย')
elif order == 'ไข่ต้ม':
    print('แนะนำให้สั่งคู่กับยำไข่ดาว')
else:
    print('ร้านเรามีไข่ดาว, ไข่เจียว, ไข่ต้ม')
```

Output :

```
Please order: ไข่เจียว
แนะนำให้สั่งคู่กับไข่ลูกเขยลูกเขย
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ order
บรรทัดที่ 2-3 : ถ้า order เป็น 'ไข่ดาว' ให้พิมพ์ 'แนะนำให้สั่งคู่กับต้มจืดไข่น้ำ'
บรรทัดที่ 4-5 : ถ้า order เป็น 'ไข่เจียว' ให้พิมพ์ 'แนะนำให้สั่งคู่กับไข่ลูกเขย'
บรรทัดที่ 6-7 : ถ้า order เป็น 'ไข่ต้ม' ให้พิมพ์ 'แนะนำให้สั่งคู่กับยำไข่ดาว'
บรรทัดที่ 8-9 : ถ้า order เป็นอื่น ๆ ให้พิมพ์ 'ร้านเรามีไข่ดาว, ไข่เจียว, ไข่ต้ม'

09 FOR & WHILE LOOP (FLOWCHART-CODE)



สแกนเพื่อเรียนวิดีโอเนื้อหาและเฉลยแบบฝึกหัดบทที่ 9

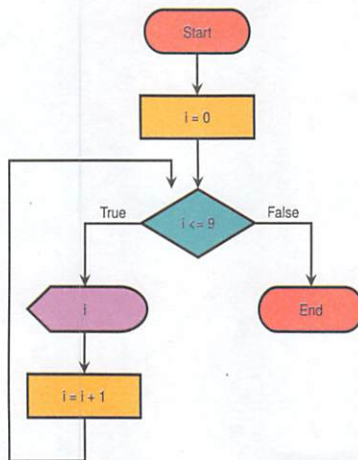
9.1 Problem

9

จงวาด flowchart จาก code ที่กำหนด

Ex. for i in range(10):

```
for i in range(10):
    print(i)
```



1.
```
for i in range(10):
    print(i)
```

2.
```
for i in range(0, 10):
    print(i)
```

3.
```
for i in range(1, 10):
    print(i)
```

4.
```
for i in range(3, 11):
    print(i)
```

5.
```
for i in range(-2, 8):
    print(i)
```

6.
```
for i in range(-7, 1):
    print(i)
```

7.
```
for i in range(-11, -5):
    print(i)
```

8.
```
for i in range(0, 11, 2):
    print(i)
```

9.
```
for i in range(0, 10, 1):
    print(i)
```

10.
```
for i in range(3, 14, 3):
    print(i)
```

11.
```
for i in range(-15, 4, 4):
    print(i)
```

12.
```
for i in range(10, 2, -1):
    print(i)
```

13.
```
for i in range(10, -1, -1):
    print(i)
```

14.
```
for i in range(7, -4, -1):
    print(i)
```

15.
```
for i in range(10, -1, -3):
    print(i)
```

16.
```
for i in range(15, -4, -5):
    print(i)
```

17.
```
for i in range(-6, 8, 2):
    for j in range(1, 9, 3):
        print(i,j)
```

9

18.
```
for i in range(0, 8):
    for j in range(6, -3, -1):
        print(i,j)
```

19.
```
for i in range(12, -1, -2):
    for j in range(-5, 1):
        print(i,j)
```

20.
```
for i in range(14, 9, -1):
    for j in range(4, -2, -1):
        print(i,j)
```

21.
```
i = 0
while i <= 10:
    print(i)
    i = i + 1
```

22.

```
i = -5
while i <= 15:
    print(i)
    i = i + 2
```

23.

```
i = 10
while i >= 1:
    print(i)
    i = i - 1
```

24.

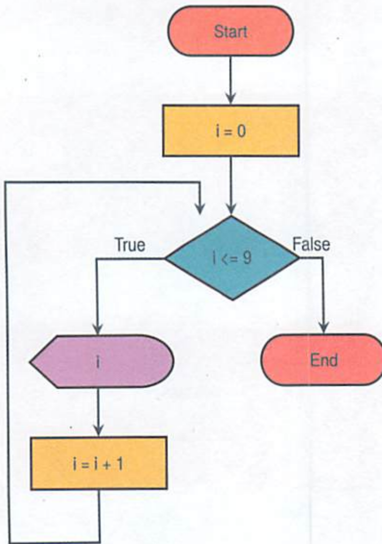
```
i = 10
while i >= -5:
    print(i)
    i = i - 3
```

25.

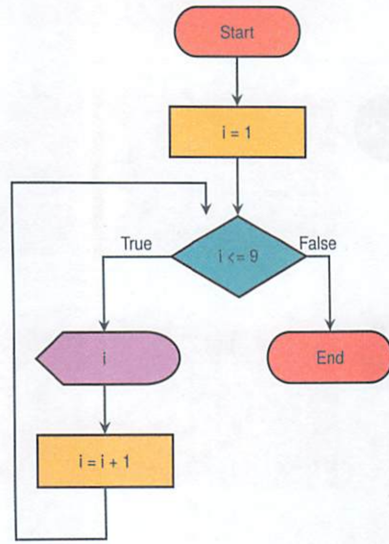
```
i = 10
while i >= 1:
    j = 1
    while j <= 5:
        print(i, j)
        j = j + 1
    i = i - 1
```

9.2 Solution

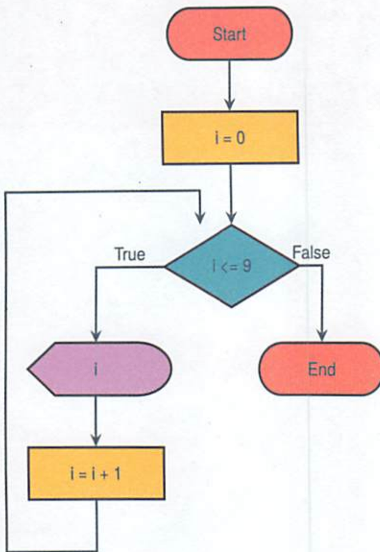
เฉลยข้อที่ 1.



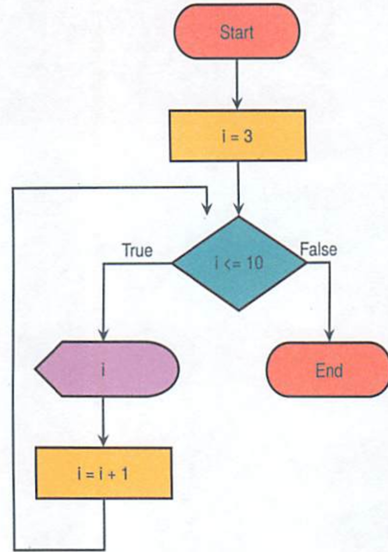
เฉลยข้อที่ 3.



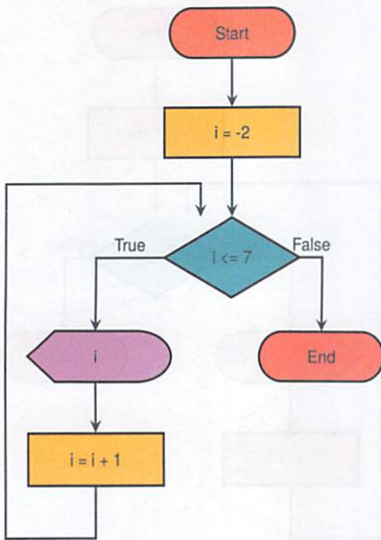
เฉลยข้อที่ 2.



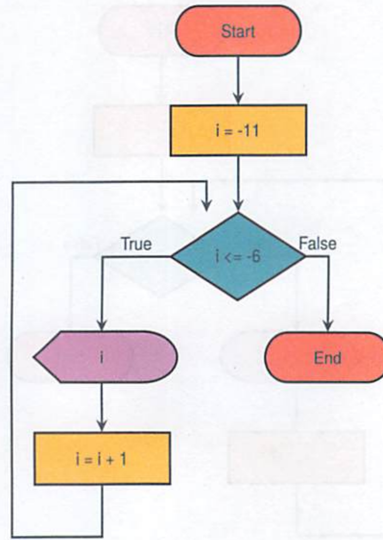
เฉลยข้อที่ 4.



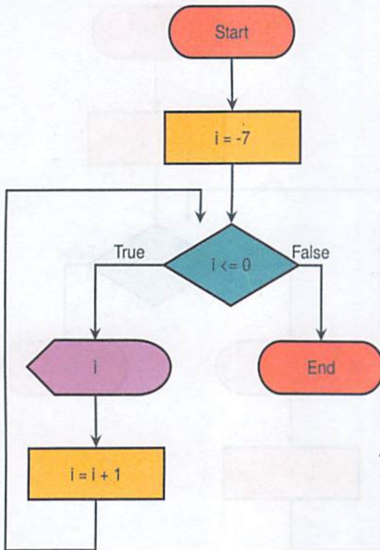
เฉลยข้อที่ 5.



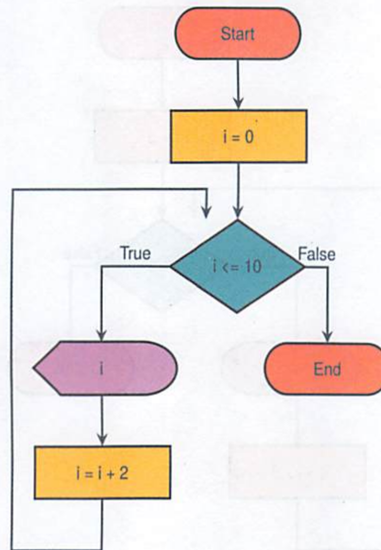
เฉลยข้อที่ 7.



เฉลยข้อที่ 6.

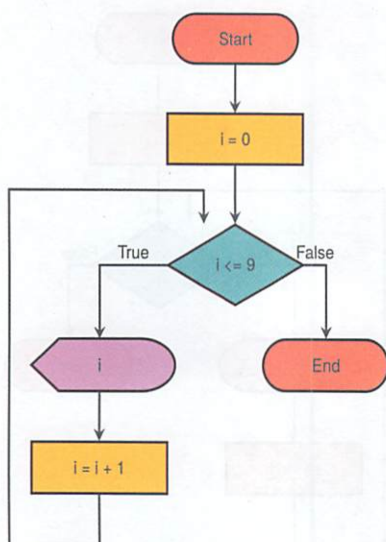


เฉลยข้อที่ 8.

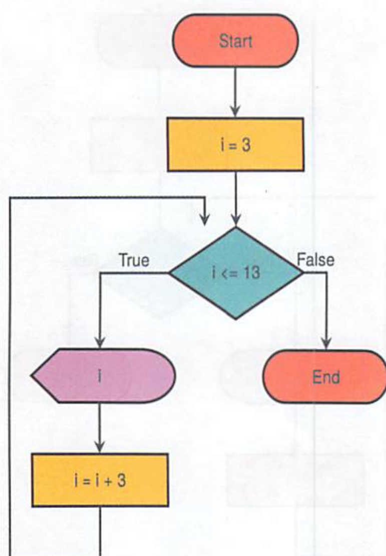


9

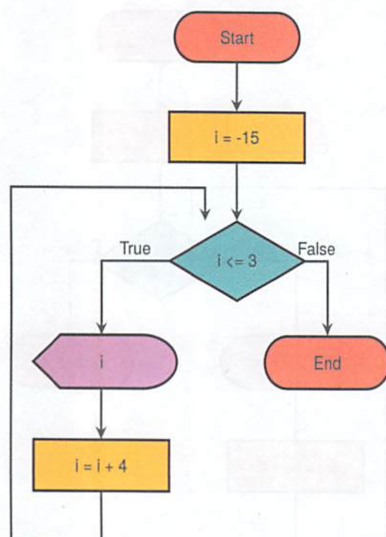
เฉลยข้อที่ 9.



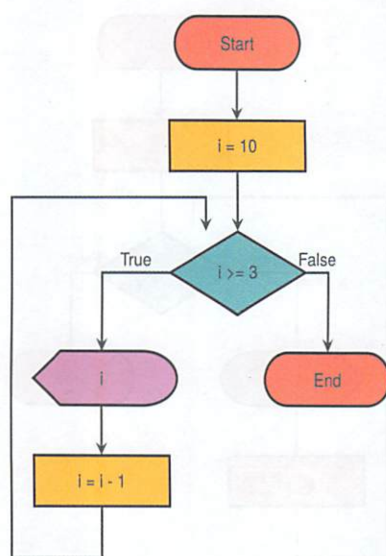
เฉลยข้อที่ 10.



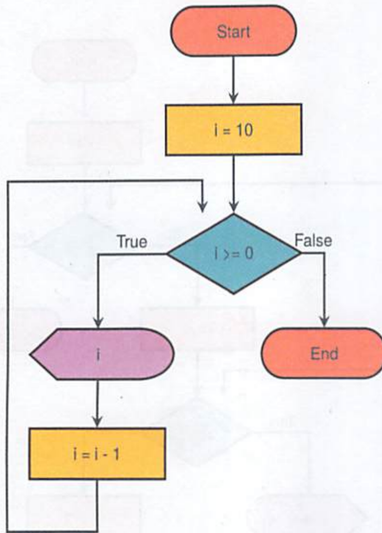
เฉลยข้อที่ 11.



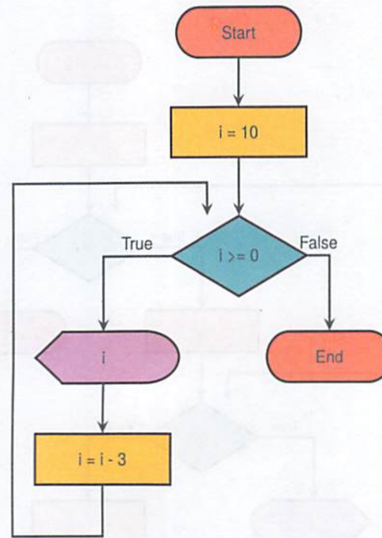
เฉลยข้อที่ 12.



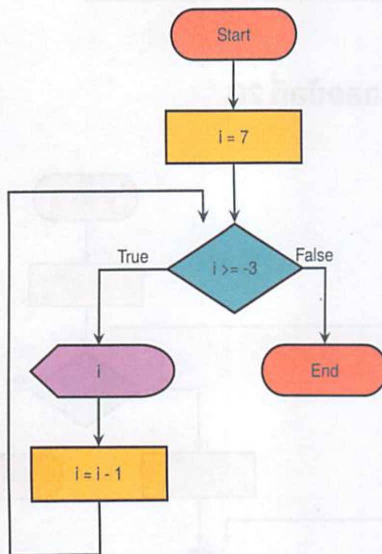
เฉลยข้อที่ 13.



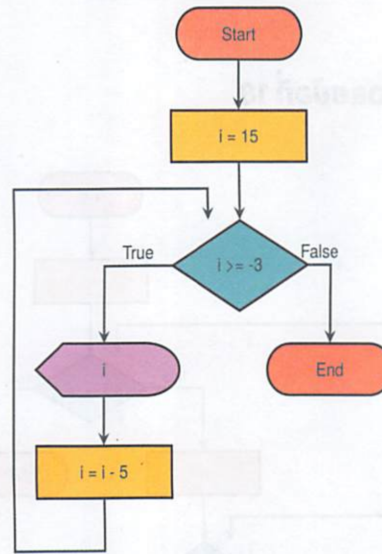
เฉลยข้อที่ 15.



เฉลยข้อที่ 14.

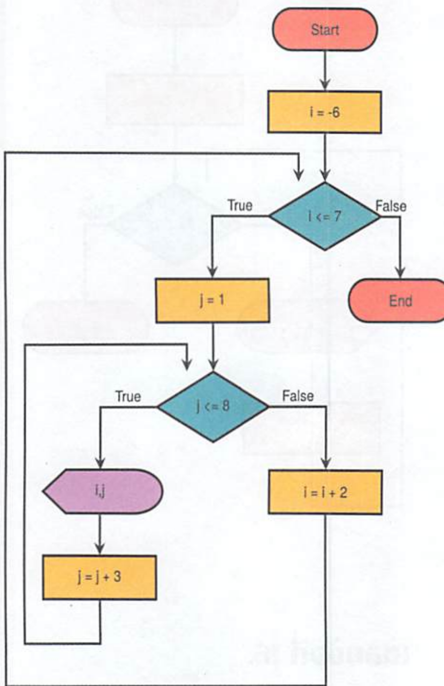


เฉลยข้อที่ 16.

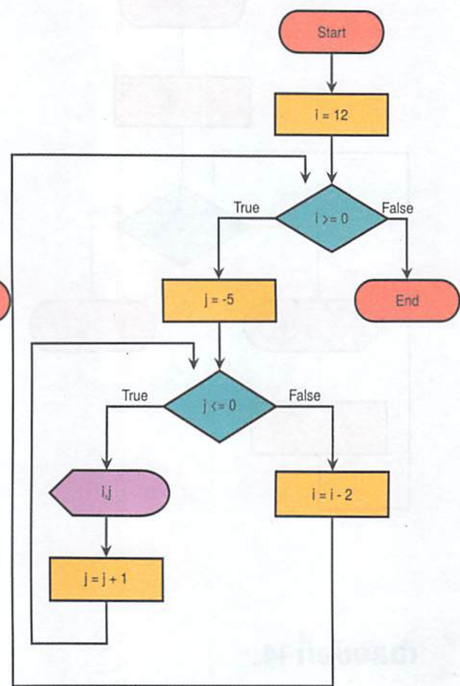


9

เฉลยข้อที่ 17.

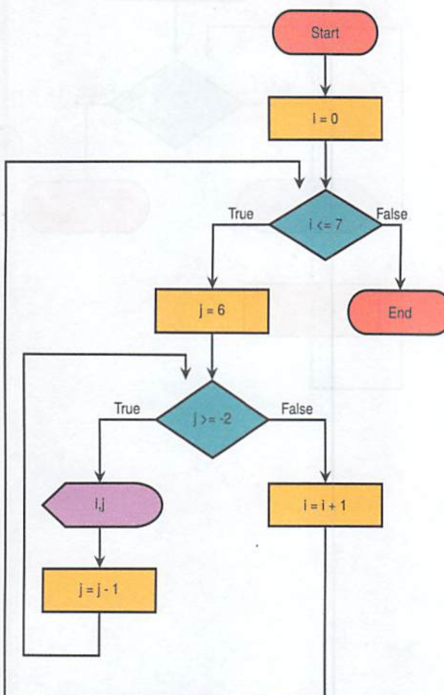


เฉลยข้อที่ 19.

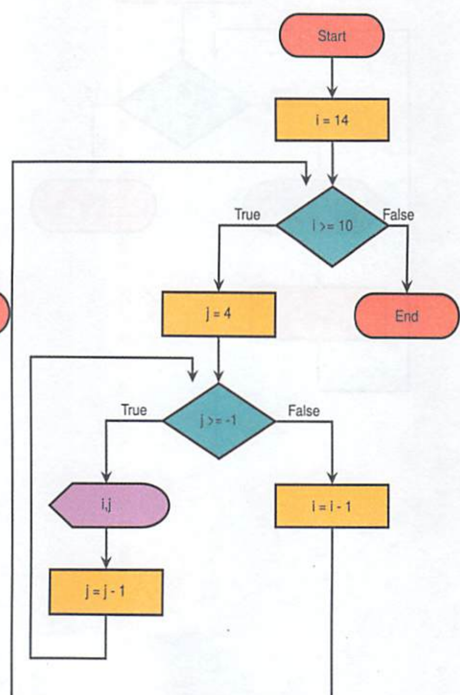


9

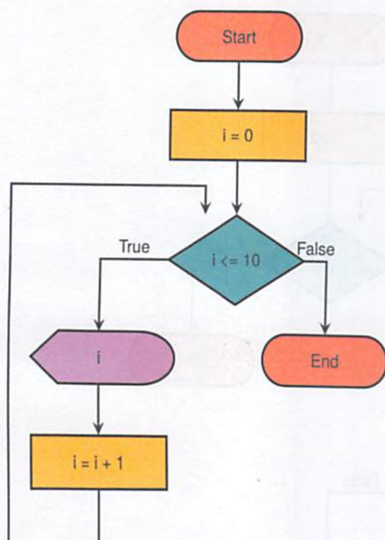
เฉลยข้อที่ 18.



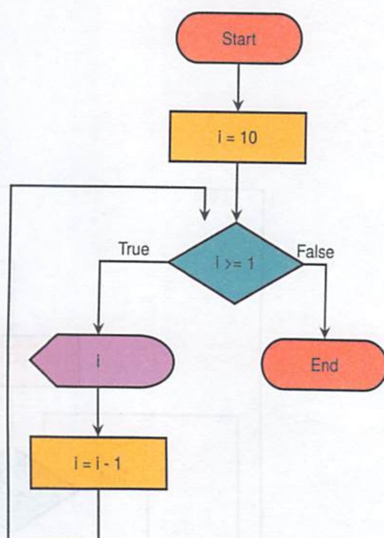
เฉลยข้อที่ 20.



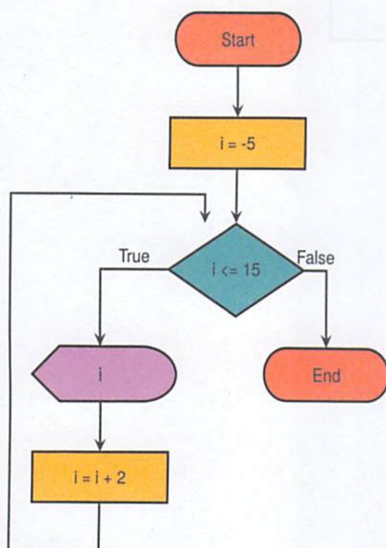
เฉลยข้อที่ 21.



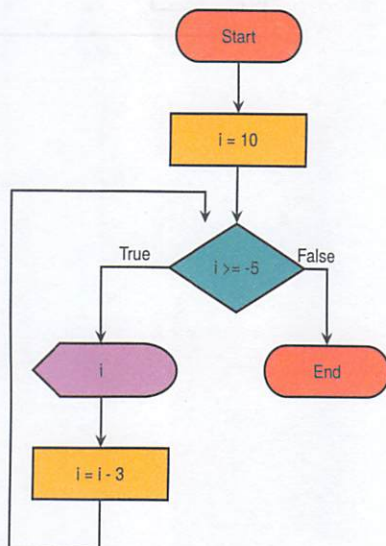
เฉลยข้อที่ 23.



เฉลยข้อที่ 22.

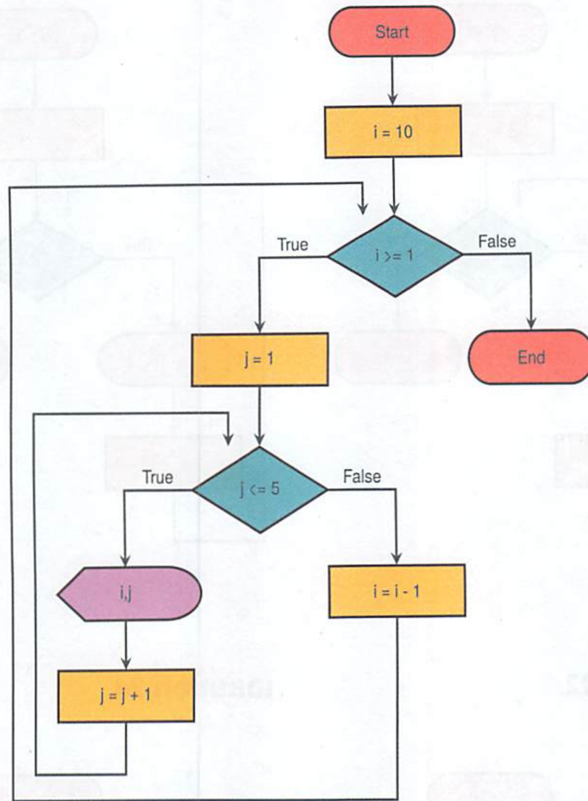


เฉลยข้อที่ 24.



9

เฉลยข้อที่ 25.



10 FOR & WHILE LOOP (CODE-FLOWCHART)

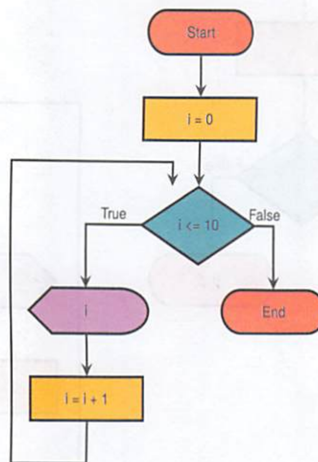


สแกนเพื่อเรียนวิดีโอเนื้อหาและเฉลยแบบฝึกหัดบทที่ 10

10.1 Problem

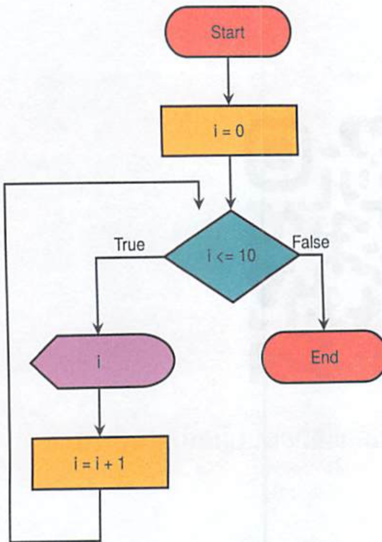
จงเขียน code จาก flowchart ที่กำหนด

Ex.

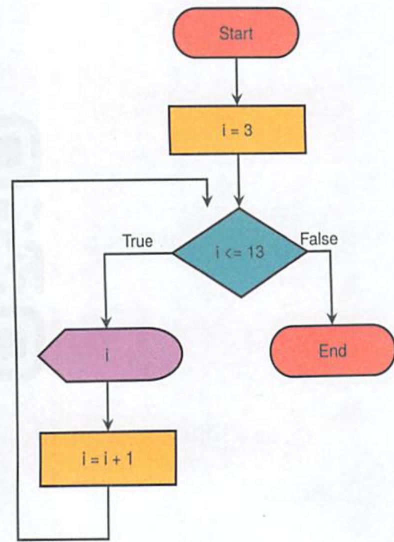



```
for i in range(11):
    print(i)
```

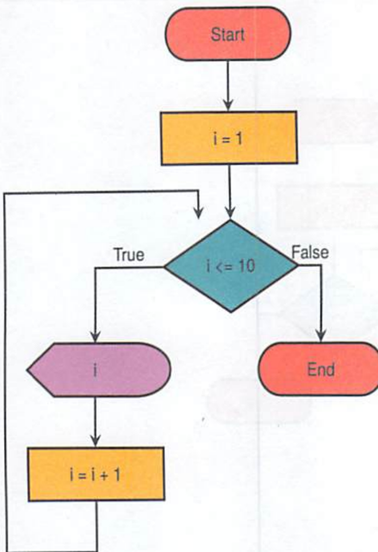
1.


Start
i = 0
i <= 10
True
False
i
End
i = i + 1

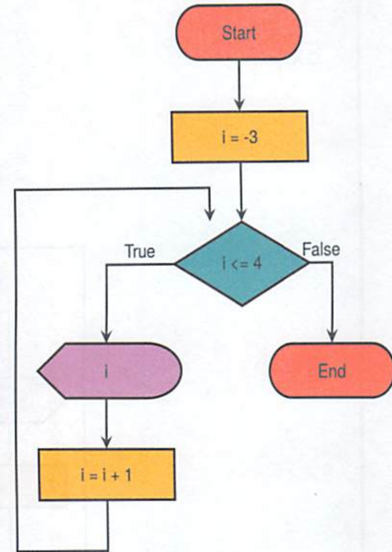
3.


Start
i = 3
i <= 13
True
False
i
End
i = i + 1

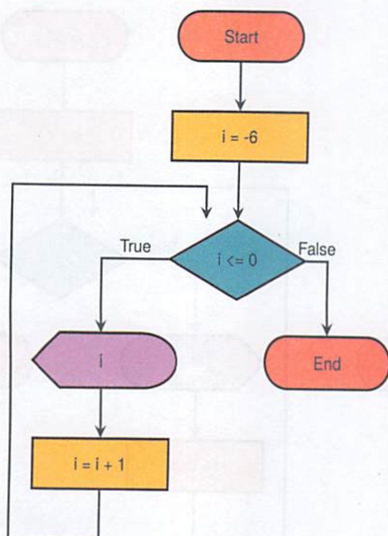
2.


Start
i = 1
i <= 10
True
False
i
End
i = i + 1

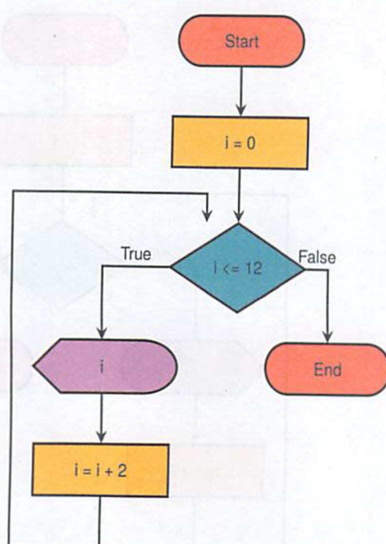
4.


Start
i = -3
i <= 4
True
False
i
End
i = i + 1

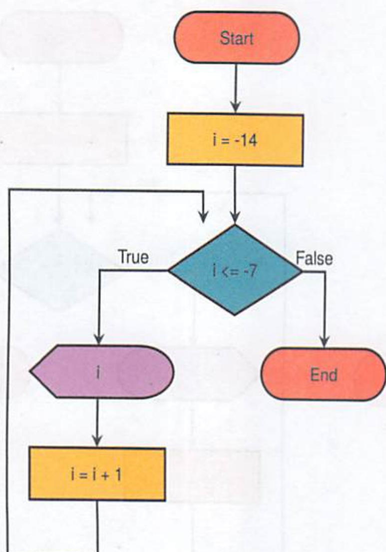
5.


Start
i = -6
i <= 0
True
False
i
End
i = i + 1

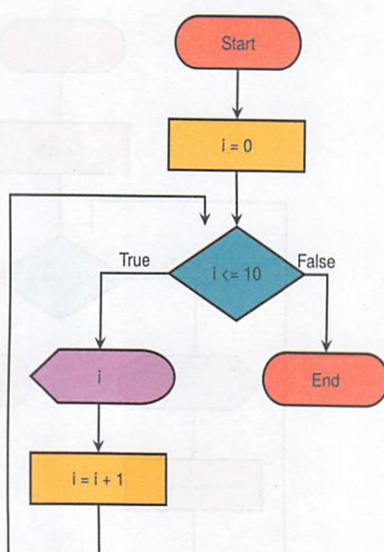
7.


Start
i = 0
i <= 12
True
False
i
End
i = i + 2

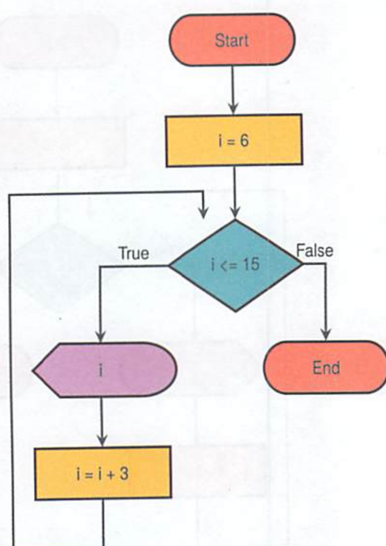
6.


Start
i = -14
i <= -7
True
False
i
End
i = i + 1

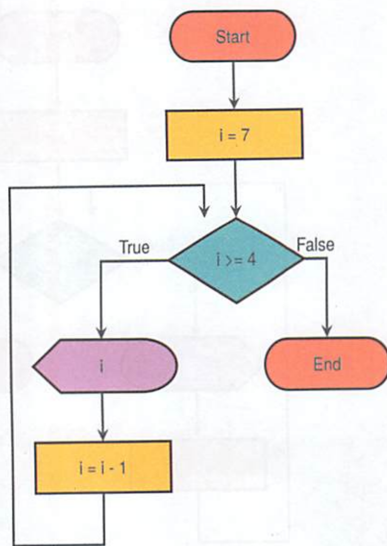
8.


Start
i = 0
i <= 10
True
False
i
End
i = i + 1

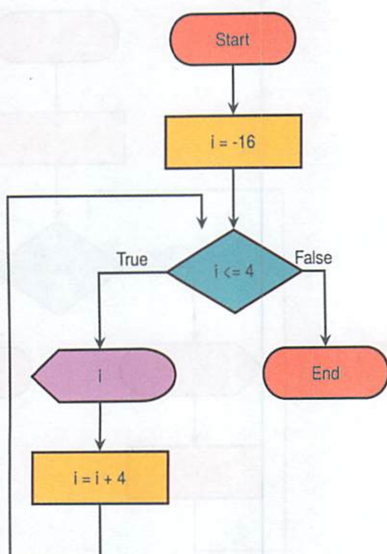
9.


Start
i = 6
True
i <= 15
False
i
End
i = i + 3

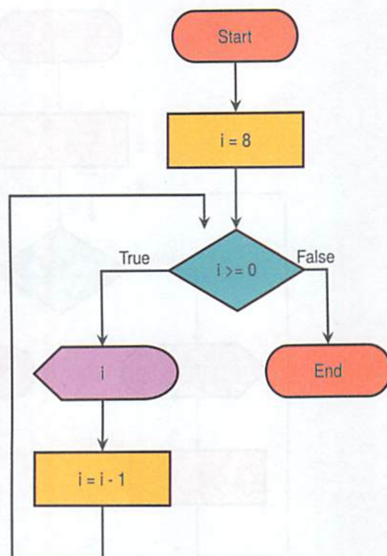
11.


Start
i = 7
True
i >= 4
False
i
End
i = i - 1

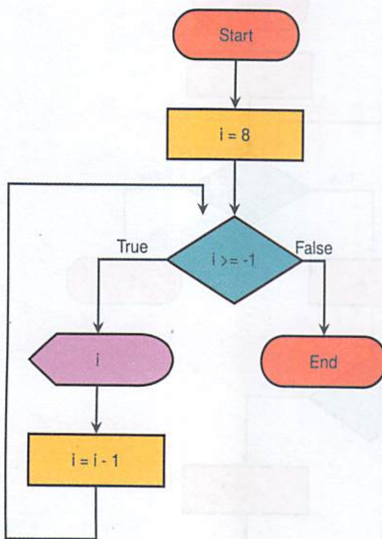
10.


Start
i = -16
True
i <= 4
False
i
End
i = i + 4

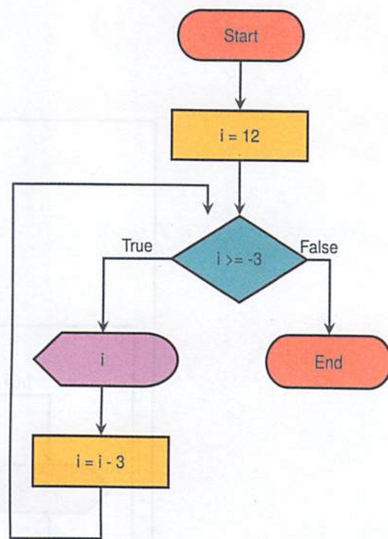
12.


Start
i = 8
True
i >= 0
False
i
End
i = i - 1

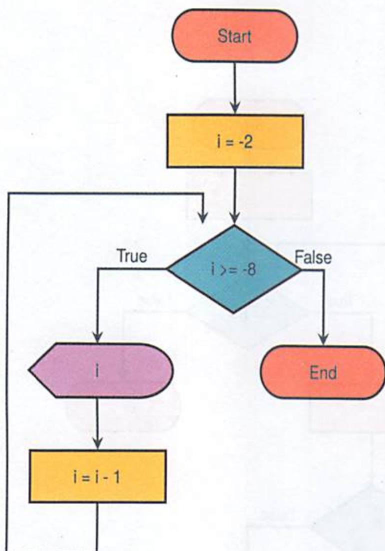
13.


Start
i = 8
i >= -1
True
False
i
End
i = i - 1

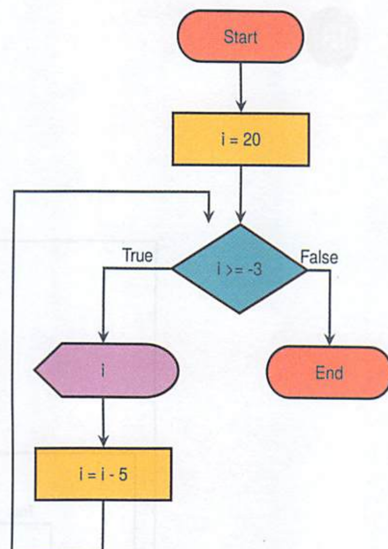
15.


Start
i = 12
i >= -3
True
False
i
End
i = i - 3

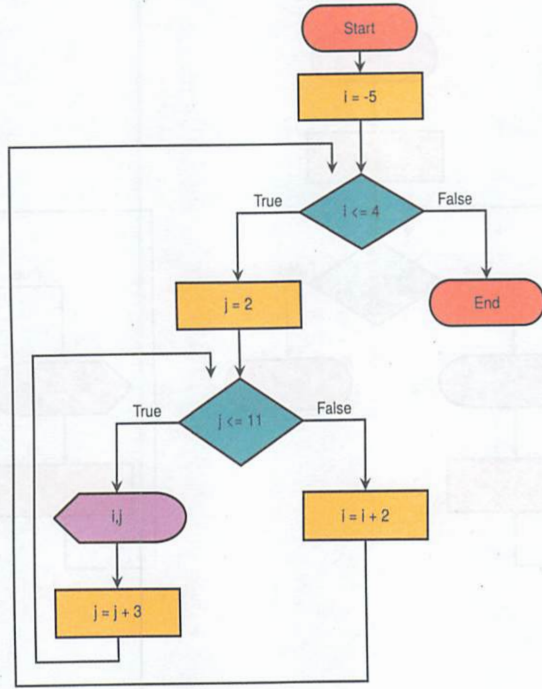
14.


Start
i = -2
i >= -8
True
False
i
End
i = i - 1

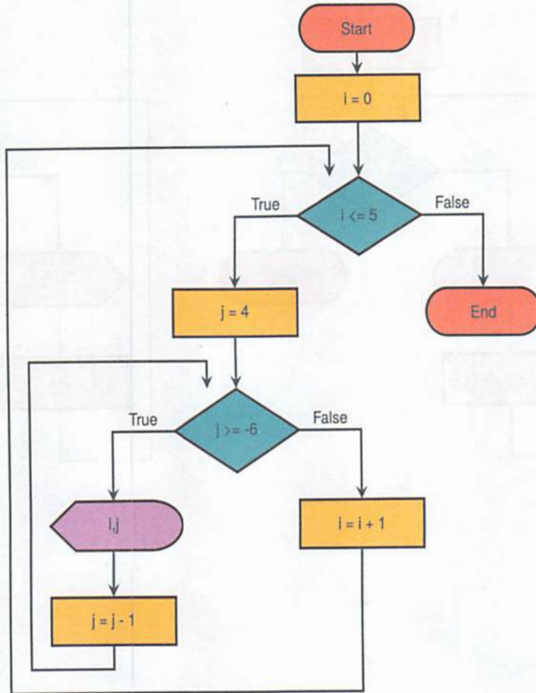
16.


Start
i = 20
i >= -3
True
False
i
End
i = i - 5

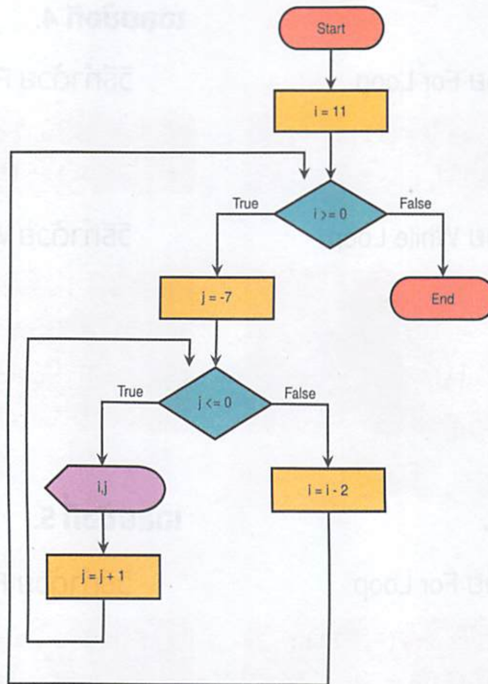
17.


Start
i = -5
True
i <= 4
False
End
j = 2
True
j <= 11
False
i,j
i = i + 2
j = j + 3

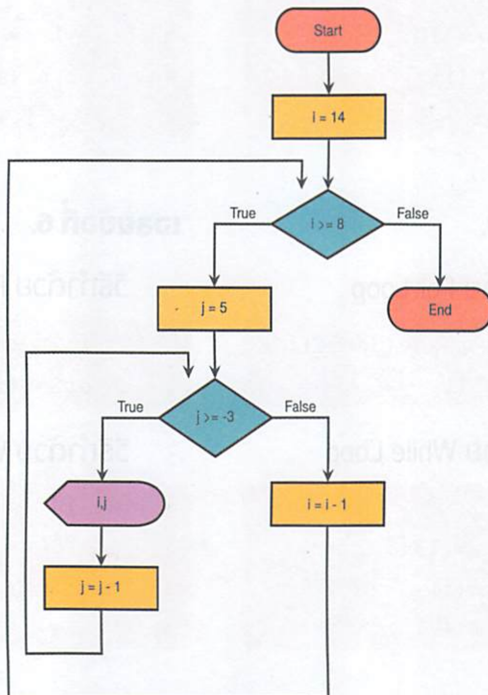
18.


Start
i = 0
True
i <= 5
False
End
j = 4
True
j >= -6
False
i,j
i = i + 1
j = j - 1

19.


Start
i = 11
True
i >= 0
False
j = -7
End
True
j <= 0
False
i,j
i = i - 2
j = j + 1

20.


Start
i = 14
True
i >= 8
False
j = 5
End
True
j >= -3
False
i,j
i = i - 1
j = j - 1

10.2 Solution

เฉลยข้อที่ 1.

วิธีทำด้วย For Loop

```
for i in range(11):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 0
while i <= 10:
    print(i)
    i = i + 1
```

เฉลยข้อที่ 2.

วิธีทำด้วย For Loop

```
for i in range(1, 11):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 1
while i <= 10:
    print(i)
    i = i + 1
```

เฉลยข้อที่ 3.

วิธีทำด้วย For Loop

```
for i in range(3, 14):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 3
while i <= 13:
    print(i)
    i = i + 1
```

เฉลยข้อที่ 4.

วิธีทำด้วย For Loop

```
for i in range(-3, 5):
    print(i)
```

วิธีทำด้วย While Loop

```
i = -3
while i <= 4:
    print(i)
    i = i + 1
```

เฉลยข้อที่ 5.

วิธีทำด้วย For Loop

```
for i in range(-6, 1):
    print(i)
```

วิธีทำด้วย While Loop

```
i = -6
while i <= 0:
    print(i)
    i = i + 1
```

เฉลยข้อที่ 6.

วิธีทำด้วย For Loop

```
for i in range(-14, -6):
    print(i)
```

วิธีทำด้วย While Loop

```
i = -14
while i <= -7:
    print(i)
    i = i + 1
```

เฉลยข้อที่ 7.

วิธีทำด้วย For Loop

```
for i in range(0, 13, 2):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 0
while i <= 12:
    print(i)
    i = i + 2
```

เฉลยข้อที่ 8.

วิธีทำด้วย For Loop

```
for i in range(0, 11, 1):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 0
while i <= 10:
    print(i)
    i = i + 1
```

เฉลยข้อที่ 9.

วิธีทำด้วย For Loop

```
for i in range(6, 16, 3):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 6
while i <= 15:
    print(i)
    i = i + 3
```

เฉลยข้อที่ 10.

วิธีทำด้วย For Loop

```
for i in range(-16, 5, 4):
    print(i)
```

วิธีทำด้วย While Loop

```
i = -16
while i <= 4:
    print(i)
    i = i + 4
```

เฉลยข้อที่ 11.

วิธีทำด้วย For Loop

```
for i in range(7, 3, -1):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 7
while i >= 4:
    print(i)
    i = i - 1
```

เฉลยข้อที่ 12.

วิธีทำด้วย For Loop

```
for i in range(8, -1, -1):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 8
while i >= 0:
    print(i)
    i = i - 1
```

เฉลยข้อที่ 13.

วิธีทำด้วย For Loop

```
for i in range(8, -2, -1):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 8
while i >= -1:
    print(i)
    i = i - 1
```

เฉลยข้อที่ 14.

วิธีทำด้วย For Loop

```
for i in range(-2, -9, -1):
    print(i)
```

วิธีทำด้วย While Loop

```
i = -2
while i >= -8:
    print(i)
    i = i - 1
```

เฉลยข้อที่ 15.

วิธีทำด้วย For Loop

```
for i in range(12, -4, -3):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 12
while i >= -3:
    print(i)
    i = i - 3
```

เฉลยข้อที่ 16.

วิธีทำด้วย For Loop

```
for i in range(20, -4, -5):
    print(i)
```

วิธีทำด้วย While Loop

```
i = 20
while i >= -3:
    print(i)
    i = i - 5
```

เฉลยข้อที่ 17.

วิธีทำด้วย For Loop

```
for i in range(-5, 5, 2):
    for j in range(2, 12, 3):
        print(i, j)
```

วิธีทำด้วย While Loop

```
i = -5
while i <= 4:
    j = 2
    while j <= 11:
        print(i, j)
        j = j + 3
    i = i + 2
```

เฉลยข้อที่ 18.

วิธีทำด้วย For Loop

```
for i in range(0, 6):
    for j in range(4, -7, -1):
        print(i, j)
```

วิธีทำด้วย While Loop

```
i = 0
while i <= 5:
    j = 4
    while j >= -6:
        print(i, j)
        j = j - 1
    i = i + 1
```

เฉลยข้อที่ 19.

วิธีทำด้วย For Loop

```
for i in range(11, -1, -2):
    for j in range(-7, 1):
        print(i, j)
```

วิธีทำด้วย While Loop

```
i = 11
while i >= 0:
    j = -7
    while j <= 0:
        print(i, j)
        j = j + 1
    i = i - 2
```

เฉลยข้อที่ 20.

วิธีทำด้วย For Loop

```
for i in range(14, 7, -1):
    for j in range(5, -4, -1):
        print(i, j)
```

วิธีทำด้วย While Loop

```
i = 14
while i >= 8:
    j = 5
    while j >= -3:
        print(i, j)
        j = j - 1
    i = i - 1
```

11 FOR & WHILE LOOP (PROBLEM)



สแกนเพื่อเรียนวิดีโอเนื้อหา
และเฉลยแบบฝึกหัดบทที่ 11



สแกนเพื่อรับ Code บทที่ 11

11.1 Problem

จงวาด flowchart และเขียน code ของโจทย์ต่อไปนี้

1. เขียนโปรแกรมพิมพ์ 0, 1, 2, ... , 19 โดยการใช้ for loop หรือ while loop
2. เขียนโปรแกรมพิมพ์ 0, 2, 4, ... , 18 โดยการใช้ for loop หรือ while loop
3. เขียนโปรแกรมพิมพ์ 1, 3, 5, ... , 19 โดยการใช้ for loop หรือ while loop
4. เขียนโปรแกรมพิมพ์ 19, 18, 17, ... , 0 โดยการใช้ for loop หรือ while loop
5. เขียนโปรแกรมพิมพ์ 18, 16, 14, ... , 0 โดยการใช้ for loop หรือ while loop
6. เขียนโปรแกรมพิมพ์ 19, 17, 15, ... , 1 โดยการใช้ for loop หรือ while loop
7. เขียนโปรแกรมพิมพ์ -3, -2, -1, 0, ... , 9 โดยการใช้ for loop หรือ while loop
8. เขียนโปรแกรมพิมพ์ -9, 8, -7, ..., 0 โดยการใช้ for loop หรือ while loop
9. เขียนโปรแกรมพิมพ์ -5, -6, -7, ... , -15 โดยการใช้ for loop หรือ while loop
10. เขียนโปรแกรมพิมพ์ -5, -7, -9, ... , -15 โดยการใช้ for loop หรือ while loop

11. เขียนโปรแกรมรับค่า n แล้วคำนวณค่าต่อไปนี้ และพิมพ์ออกมา

$$1 + 2 + 3 + \ldots + n$$

12. เขียนโปรแกรมรับค่า n แล้วคำนวณค่าต่อไปนี้ และพิมพ์ออกมา

$$1 \times 2 \times 3 \times \ldots \times n$$

13. เขียนโปรแกรมรับค่า n แล้วคำนวณค่าต่อไปนี้ และพิมพ์ออกมา

$$\frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \ldots + \frac{1}{n}$$

14. เขียนโปรแกรมรับค่า n แล้วคำนวณค่าต่อไปนี้ และพิมพ์ออกมา

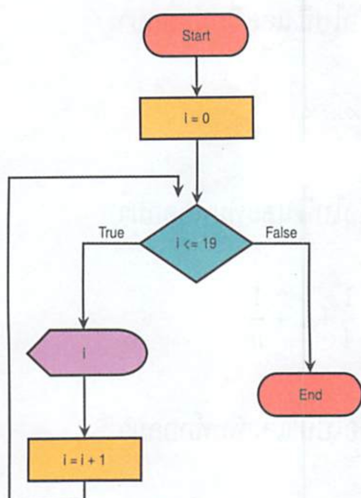
$$\frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \ldots + \frac{1}{n^2}$$

15. เขียนโปรแกรมรับค่า n แล้วคำนวณค่าต่อไปนี้ และพิมพ์ออกมา

$$\sqrt{6 \times (\frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \ldots + \frac{1}{n^2})}$$

11.2 Solution

เฉลยข้อที่ 1.



วิธีทำด้วย For Loop

Input :

```
for i in range(20):
    print(i)
```

วิธีทำด้วย While Loop

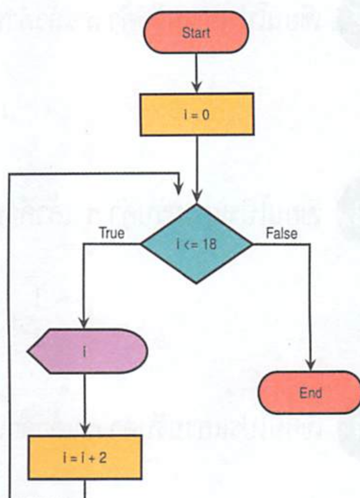
Input :

```
i = 0
while i <= 19:
    print(i)
    i = i + 1
```

Output :

```
1
2
3
⋮
20
```

เฉลยข้อที่ 2.



วิธีทำด้วย For Loop

Input :

```
for i in range(0, 19, 2):
    print(i)
```

วิธีทำด้วย While Loop

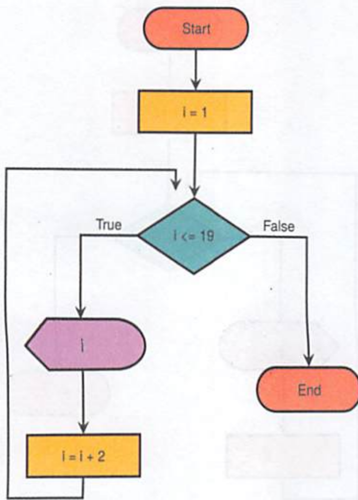
Input :

```
i = 0
while i <= 18:
    print(i)
    i = i + 2
```

Output :

```
0
2
4
⋮
18
```

เฉลยข้อที่ 3.



วิธีทำด้วย For Loop

Input :

```
for i in range(1, 20, 2):
    print(i)
```

วิธีทำด้วย While Loop

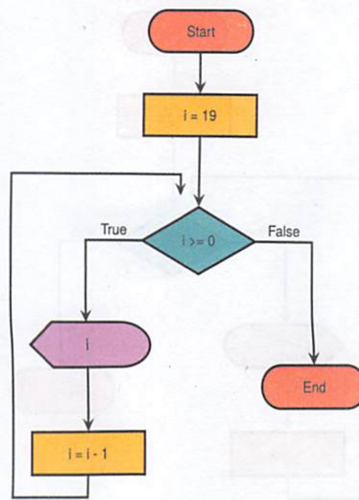
Input :

```
i = 1
while i <= 19:
    print(i)
    i = i + 2
```

Output :

```
1
3
5
:
19
```

เฉลยข้อที่ 4.



วิธีทำด้วย For Loop

Input :

```
for i in range(19, -1, -1)
 ↪ :
    print(i)
```

วิธีทำด้วย While Loop

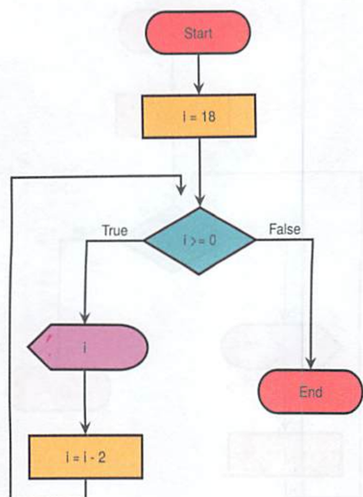
Input :

```
i = 19
while i >= 0:
    print(i)
    i = i - 1
```

Output :

```
19
18
17
:
0
```

เฉลยข้อที่ 5.



วิธีทำด้วย For Loop

Input :

```
for i in range(18, -1, -2)
    ↪ :
    print(i)
```

วิธีทำด้วย While Loop

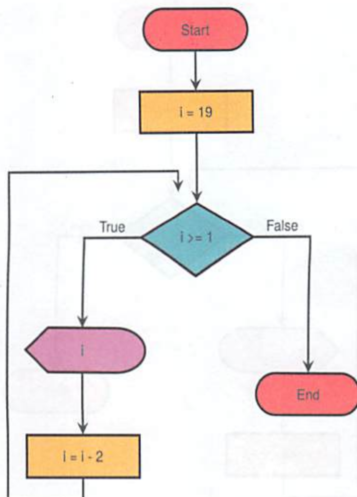
Input :

```
i = 18
while i >= 0:
    print(i)
    i = i - 2
```

Output :

```
18
16
14
:
0
```

เฉลยข้อที่ 6.



วิธีทำด้วย For Loop

Input :

```
for i in range(19, 0, -2):
    print(i)
```

วิธีทำด้วย While Loop

Input :

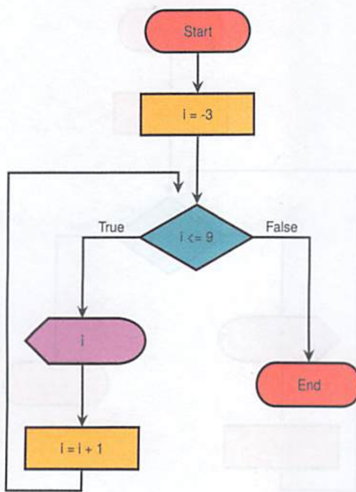
```
i = 19
while i >= 1:
    print(i)
    i = i - 2
```

Output :

```
19
17
15
:
1
```

11

เฉลยข้อที่ 7.



วิธีทำด้วย For Loop

Input :

```
for i in range(-3, 10):
    print(i)
```

วิธีทำด้วย While Loop

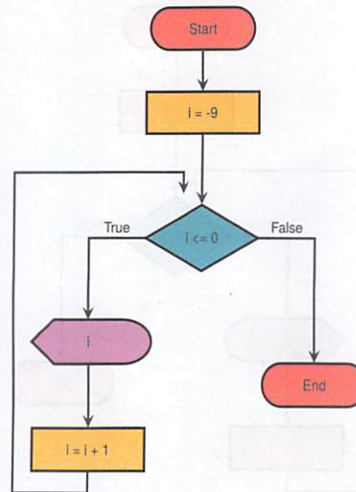
Input :

```
i = -3
while i <= 9:
    print(i)
    i = i + 1
```

Output :

```
-3
-2
-1
:
9
```

เฉลยข้อที่ 8.



วิธีทำด้วย For Loop

Input :

```
for i in range(-9, 1):
    print(i)
```

วิธีทำด้วย While Loop

Input :

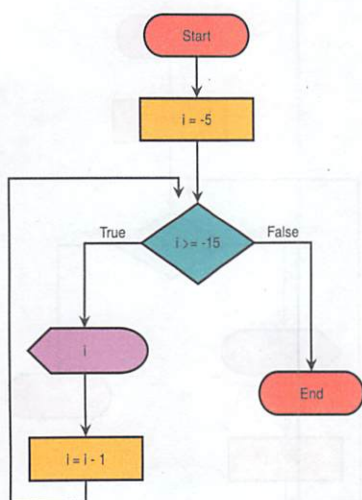
```
i = -9
while i <= 0:
    print(i)
    i = i + 1
```

Output :

```
-9
-8
-7
:
0
```

11

เฉลยข้อที่ 9.



วิธีทำด้วย For Loop

Input :

```
for i in range(-5, -16,
    ↪ -1):
    print(i)
```

วิธีทำด้วย While Loop

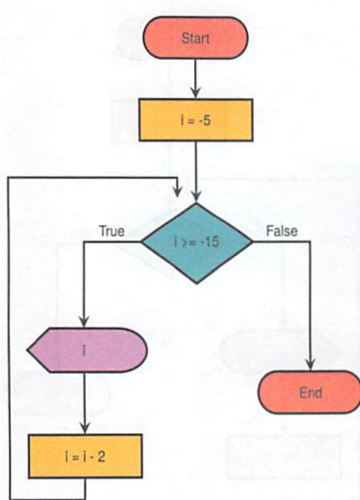
Input :

```
i = -5
while i >= -15:
    print(i)
    i = i - 1
```

Output :

```
-5
-6
-7
⋮
-15
```

เฉลยข้อที่ 10.



วิธีทำด้วย For Loop

Input :

```
for i in range(-5, -16,
    ↪ -2):
    print(i)
```

วิธีทำด้วย While Loop

Input :

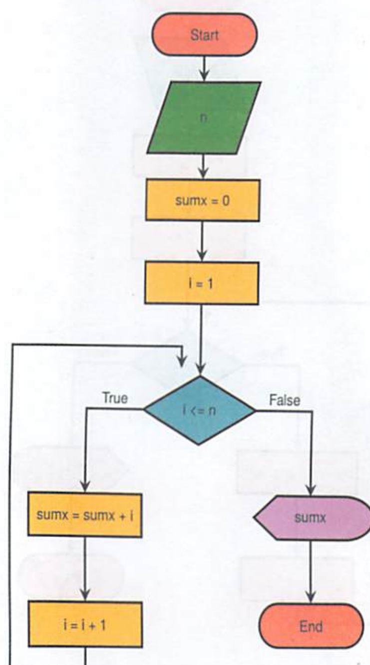
```
i = -5
while i >= -15:
    print(i)
    i = i - 2
```

Output :

```
-5
-7
-9
⋮
-15
```

11

เฉลยข้อที่ 11.



วิธีทำด้วย For Loop

Input :

```
n = int(input('Please insert number : '))
sumx = 0
for i in range(1, n+1):
    sumx = sumx + i
print(sumx)
```

วิธีทำด้วย While Loop

Input :

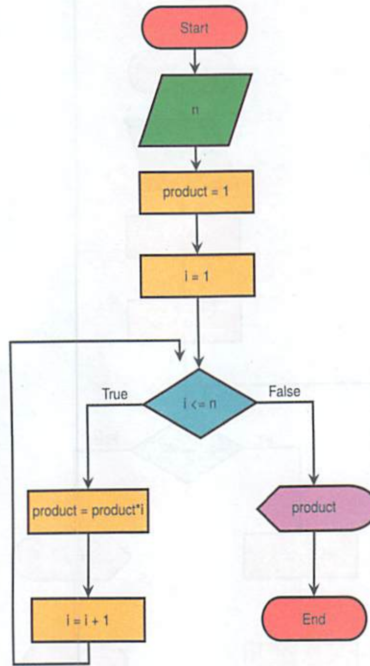
```
n = int(input('Please insert number : '))
sumx = 0
i = 1
while i <= n:
    sumx = sumx + i
    i = i + 1
print(sumx)
```

Output :

```
Please insert number : 10
55
```

11

เฉลยข้อที่ 12.



วิธีทำด้วย For Loop

Input :

```
n = int(input('Please insert number : '))
product = 1
for i in range(1, n+1):
    product = product*i
print(product)
```

วิธีทำด้วย While Loop

Input :

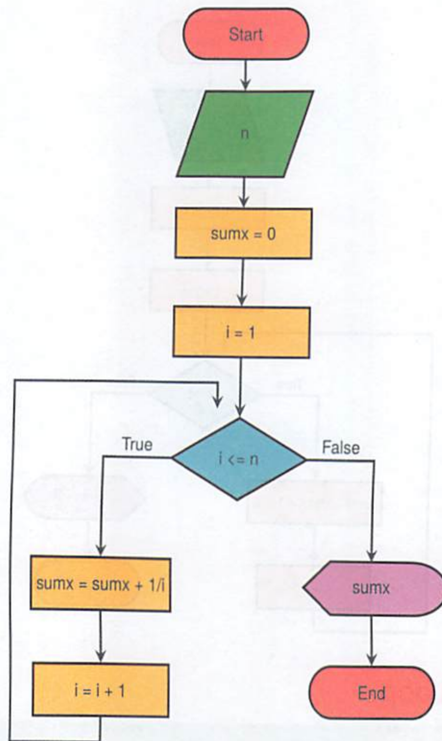
```
n = int(input('Please insert number : '))
product = 1
i = 1
while i <= n:
    product = product*i
    i = i + 1
print(product)
```

Output :

```
Please insert number : 5
120
```

11

เฉลยข้อที่ 13.



วิธีทำด้วย For Loop

Input :

```
n = int(input('Please insert number : '))
sumx = 0
for i in range(1, n+1):
    sumx = sumx + 1/i
print(sumx)
```

วิธีทำด้วย While Loop

Input :

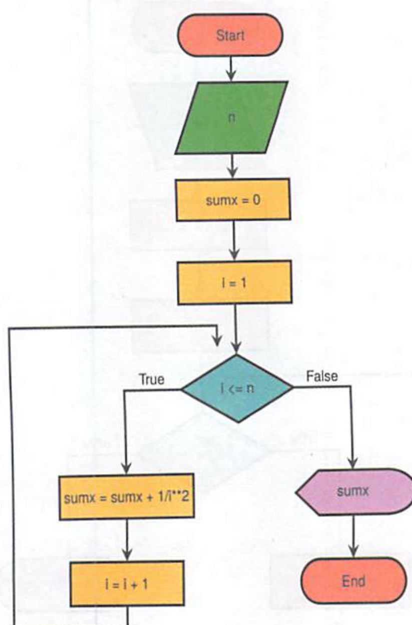
```
n = int(input('Please insert number : '))
sumx = 0
i = 1
while i <= n:
    sumx = sumx + 1/i
    i = i + 1
print(sumx)
```

Output :

```
Please insert number : 10
2.9289682539682538
```

11

เฉลยข้อที่ 14.



วิธีทำด้วย For Loop

Input :

```
n = int(input('Please insert number : '))
sumx = 0
for i in range(1, n+1):
    sumx = sumx + 1/i**2
print(sumx)
```

วิธีทำด้วย While Loop

Input :

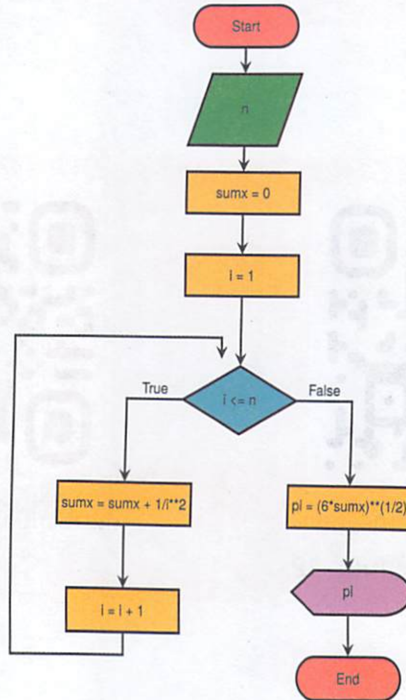
```
n = int(input('Please insert number : '))
sumx = 0
i = 1
while i <= n:
    sumx = sumx + 1/i**2
    i = i + 1
print(sumx)
```

Output :

```
Please insert number : 10
1.5497677311665408
```

11

เฉลยข้อที่ 15.



วิธีทำด้วย For Loop

Input :

```
n = int(input('Please insert number : '))
sumx = 0
for i in range(1, n+1):
    sumx = sumx + 1/i**2
pi = (6*sumx)**(1/2)
print(pi)
```

วิธีทำด้วย While Loop

Input :

```
n = int(input('Please insert number : '))
sumx = 0
i = 1
while i <= n:
    sumx = sumx + 1/i**2
    i = i + 1
pi = (6*sumx)**(1/2)
print(pi)
```

Output :

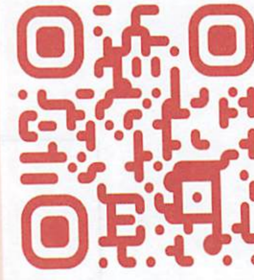
```
Please insert number : 100
3.1320765318091053
```

11

12 PROBLEM



สแกนเพื่อเรียนวิดีโอเนื้อหา และเฉลยแบบฝึกหัดบทที่ 12



สแกนเพื่อรับ Code บทที่ 12

12.1 Problem

จงวาด flowchart และเขียน code ของโจทย์ต่อไปนี้

12

1. เขียนโปรแกรมพิมพ์จำนวนที่หารด้วย 3 ลงตัว ซึ่งอยู่ระหว่าง 0 และ 100
2. เขียนโปรแกรมพิมพ์จำนวนที่หารด้วย 3 และ 5 ลงตัว ซึ่งอยู่ระหว่าง 0 และ 100
3. เขียนโปรแกรมพิมพ์จำนวนที่หารด้วย 3 หรือ 5 ลงตัว ซึ่งอยู่ระหว่าง 0 และ 100
4. เขียนโปรแกรมพิมพ์จำนวนที่หารด้วย 3, 5 และ 7 ลงตัว ซึ่งอยู่ระหว่าง 0 และ 1000
5. เขียนโปรแกรมพิมพ์จำนวนที่หารด้วย 3, 5 หรือ 7 ลงตัว ซึ่งอยู่ระหว่าง 0 และ 100
6. เขียนโปรแกรมพิมพ์จำนวนที่หารด้วย 3 หรือ 5 ลงตัวเพียงตัวใดตัวหนึ่ง ซึ่งอยู่ระหว่าง 0 และ 100
7. เขียนโปรแกรมพิมพ์จำนวนที่หารด้วย 3 และ 5 ลงตัว แต่หารด้วย 2 ไม่ลงตัว ซึ่งอยู่ระหว่าง 0 และ 100

8. เขียนโปรแกรมพิมพ์จำนวนที่หารด้วย 3 หรือ 5 ลงตัว แต่หารด้วย 2 ไม่ลงตัว ซึ่งอยู่ระหว่าง 0 และ 100

9. เขียนโปรแกรมนับจำนวนทั้งหมดที่หารด้วย 3 ลงตัว ซึ่งอยู่ระหว่าง 0 และ 1000 จากนั้นให้พิมพ์ออกมา

10. เขียนโปรแกรมหาผลบวกของจำนวนที่หารด้วย 3 ลงตัว ซึ่งอยู่ระหว่าง 0 และ 1000 ว่ามีค่าเท่าไร จากนั้นให้พิมพ์ออกมา

11. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนเต็มโดยใช้ for loop จากนั้นให้หาผลรวมของจำนวนที่รับและพิมพ์ออกมา

12. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนจริงโดยใช้ for loop จากนั้นให้หาผลรวมของจำนวนที่เป็นบวกและจำนวนที่เป็นลบ สุดท้ายให้พิมพ์ผลลัพธ์ออกมา

13. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนจริงโดยใช้ for loop จากนั้นให้หาผลคูณของจำนวนที่เป็นบวกและพิมพ์ผลลัพธ์ออกมา

14. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็มบวก หลังจากนั้นให้ทำการตรวจสอบว่า มีจำนวนเต็มบวกตัวใดบ้างที่หารจำนวนที่รับมาลงตัวและพิมพ์จำนวนเหล่านั้นออกมา

15. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็มบวก หลังจากนั้นให้นับจำนวนเต็มบวกที่หารจำนวนที่รับมาลงตัวและพิมพ์ออกมา

16. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็มบวก หลังจากนั้นให้ตรวจสอบว่าจำนวนที่รับมาเป็นจำนวนเฉพาะหรือไม่
    - ถ้าเป็น ให้พิมพ์ 'is prime'
    - ถ้าไม่เป็น ให้พิมพ์ 'is not prime'

17. เขียนโปรแกรมหาจำนวนเฉพาะที่อยู่ระหว่าง 0 และ 100 และพิมพ์ออกมา

18. เขียนโปรแกรมหาจำนวนเฉพาะ 100 ตัวแรก และพิมพ์ออกมา

19. เขียนโปรแกรมหาผลบวกของจำนวนเฉพาะ 100 ตัวแรก และพิมพ์ออกมา

20. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ จากนั้นให้พิมพ์อักขระที่ index เป็นเลขคู่

12

21. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ จากนั้นให้พิมพ์อักขระที่ index เป็นเลขคี่

22. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นสายอักขระโดยใช้ for loop จากนั้นให้นับความยาวรวมของสายอักขระที่รับและพิมพ์ออกมา

23. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นสายอักขระ (str1, char1) ให้ตรวจสอบความยาวของ char1 ว่าเป็น 1 หรือไม่

    - ถ้าความยาวเป็น 1 ให้ตรวจสอบว่า char1 ปรากฎที่ index ใดบ้างของ str1 และพิมพ์ออกมา
    - ถ้าความยาวไม่เป็น 1 ไม่ดำเนินการใด ๆ

24. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นสายอักขระโดยใช้ for loop จากนั้นให้ตรวจสอบว่าในสายอักขระที่รับมาทั้งหมดมี 'a' ปรากฎกี่ครั้งและพิมพ์ออกมา

25. กำหนดให้ vowel = ['a', 'e', 'i', 'o', 'u'] เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ จากนั้นให้ตรวจสอบว่าสายอักขระที่รับมีสมาชิกใน vowel ปรากฎหรือไม่

    - ถ้ามี ให้พิมพ์ 'There is vowel'
    - ถ้าไม่มี ให้พิมพ์ 'There is no vowel'

12

26. กำหนดให้ vowel = ['a', 'e', 'i', 'o', 'u'] เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ (str1) ให้แทนที่ค่าอักขระในตำแหน่งที่สมาชิกใน vowel ปรากฏด้วย '*' จากนั้นให้พิมพ์สายอักขระที่ผ่านกระบวนการดังกล่าวออกมา

    ตัวอย่างที่ 1 : สมมติให้ str1 = 'apple' ผลลัพธ์ที่จะพิมพ์ออกมาคือ *ppl*

27. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ (str1) จากนั้นให้สร้างสายอักขระถอยหลังของอินพุตที่รับและพิมพ์ออกมา

    ตัวอย่างที่ 1 : สมมติให้ str1 = 'banana' ผลลัพธ์ที่จะพิมพ์ออกมาคือ ananab

28. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นสายอักขระโดยใช้ for loop จากนั้นให้ทำการต่อสายอักขระทั้งหมดที่รับและพิมพ์ออกมา

ตัวอย่างที่ 1 : สมมติให้สายอักขระทั้ง 5 ตัว คือ 'i', 'abc', 'three', 'on' และ 'xyz' ผลลัพธ์ที่จะพิมพ์ออกมาคือ 'iabcthreeonxyz'

29. เขียนโปรแกรมรับอินพุต 2 ตัวที่เป็นสายอักขระและจำนวนเต็มตามลำดับ (str1, int1) ถ้าความยาวของ str1 น้อยกว่า int1 ให้ทำการต่อ str1 ด้วย '*' จนกว่าความยาวของ str1 จะไม่น้อยกว่า int1 สุดท้ายให้พิมพ์ค่า str1 ที่ผ่านกระบวนการดังกล่าวออกมา

ตัวอย่างที่ 1 : สมมติให้ str1 = 'be', int = 5 ผลลัพธ์ที่จะพิมพ์ออกมาคือ be***

30. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ (str1) จากนั้นให้ทำการหาอักขระที่แตกต่างกันทั้งหมดและพิมพ์ออกมา

ตัวอย่างที่ 1 : สมมติให้ str1 = 'green' ผลลัพธ์ที่จะพิมพ์ออกมาคือ {'g', 'r' 'e' 'n' }

31. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ จากนั้นให้ทำการหาอักขระที่แตกต่างกันทั้งหมดพร้อมทั้งจำนวนครั้งที่อักขระนั้น ๆ ปรากฏและพิมพ์ออกมา

32. เขียนโปรแกรมสร้าง empty list และ append ค่า 0, 1, 2, ..., 19 ลงไปใน list นั้น หลังจากนั้นให้พิมพ์ค่า list ที่สร้างออกมา

33. เขียนโปรแกรมสร้าง empty list และ append ค่า 99, 98, 97, ..., 80 ลงไปใน list นั้น หลังจากนั้นให้พิมพ์ค่า list ที่สร้างออกมา

12

34. เขียนโปรแกรมสร้าง empty list (list1) และรับอินพุต 5 ตัวที่เป็นจำนวนเต็ม โดยใช้ for loop และให้ตรวจสอบว่า จำนวนที่รับยังไม่เป็นสมาชิกของ list1 หรือไม่

- ถ้ายังไม่เป็นสมาชิก ให้ append จำนวนนั้นเข้าไปใน list1
- ถ้าเป็นสมาชิกอยู่แล้ว ไม่ดำเนินการใด ๆ

จากนั้นให้พิมพ์ค่า list1 ออกมา

35. เขียนโปรแกรมสร้าง empty list และรับอินพุตที่เป็นสายอักขระและ append เข้าไปใน list จนกว่าความยาวของ list จะเท่ากับ 5 โดยมีเงื่อนไขว่าถ้าสายอักขระที่รับเป็นสมาชิกใน list อยู่แล้วจะไม่ทำการ append

36. กำหนดให้ list1 = [1, 2, 3, ... , 10] เขียนโปรแกรมสร้าง list ถอยหลังของ list1 และพิมพ์ออกมา

> **ตัวอย่างที่ 1** : กำหนดให้ list1 = [1, 2, 3, 4] ผลลัพธ์ที่จะพิมพ์ออกมาคือ [4, 3, 2, 1]

37. กำหนดให้ list1 = [1, 2, 3, ... , 10] เขียนโปรแกรมพิมพ์สมาชิกใน list1 ที่มี index เป็นเลขคู่

12

38. กำหนดให้ list1 = [1, 2, 3, ... , 10] เขียนโปรแกรมพิมพ์สมาชิกใน list1 ที่มี index เป็นเลขคี่

39. กำหนดให้ list1 = [1, 2, 3, ... , 10] เขียนโปรแกรมพิมพ์สมาชิกใน list1 ที่ถูกหารด้วย 3 ลงตัว

40. กำหนดให้ list2 = ['a', 'l', 'l', 'i', 'g', 'a', 't', 'o', 'r'] เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ ให้ตรวจสอบว่าสายอักขระที่รับมามีค่าตรงกับสมาชิกตำแหน่งใดใน list2 และพิมพ์ออกมา

41. กำหนดให้ list3 = [1, 2, 3, ... , 25] เขียนโปรแกรมเปลี่ยนสมาชิกใน list3 ที่ index เป็นเลขคู่ให้เป็น 'a' จากนั้นให้พิมพ์ค่า list3 ออกมา

42. กำหนดให้ list3 = [1, 2, 3, ... , 25] เขียนโปรแกรมเปลี่ยนสมาชิกใน list3 ที่ index เป็นเลขคี่ให้เป็น 'b' จากนั้นให้พิมพ์ค่า list3 ออกมา

43. กำหนดให้ list3 = [1, 2, 3, ... , 25] เขียนโปรแกรมเปลี่ยนสมาชิกใน list3 ที่ค่าสามารถถอดรากที่สองแล้วเป็นจำนวนเต็มให้เป็น 'square' จากนั้นให้พิมพ์ค่า list3 ออกมา

44. กำหนดให้ listx = [0, 2 ,4 ,6, 8] และ list4 = ['a', 'b', 'c', 'd', 'e', 'f'] เขียนโปรแกรมแทรก '*' ไปใน list4 ใน index ที่เป็นสมาชิกของ listx

45. กำหนดให้ list4 = ['a', 'b', 'c', 'd', 'e', 'f'] เขียนโปรแกรมรับ index และ element ที่จะแทรกใน list จำนวน 3 ครั้งโดยใช้ for loop จากนั้นให้พิมพ์ค่า list4 ที่ผ่านกระบวนการดังกล่าวออกมา

46. กำหนดให้ list4 = ['a', 'b', 'c', 'd', 'e', 'f'] เขียนโปรแกรมรับ index และ element ที่จะแทรกใน list จำนวน 3 ครั้งโดยใช้ for loop โดยมีเงื่อนไขว่าจะทำการแทรกก็ต่อเมื่อ element ยังไม่เป็นสมาชิกของ list4 เท่านั้น จากนั้นให้พิมพ์ค่า list4 ที่ผ่านกระบวนการดังกล่าวออกมา

47. กำหนดให้ list4 = ['a', 'b', 'c', 'd', 'e', 'f'] เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม (num1) จากนั้นให้ตรวจสอบว่า num1 มีค่าไม่เกิน index สุดท้ายของ list4 หรือไม่

- ถ้าไม่เกิน ให้ลบสมาชิกใน index ที่ num1 โดยใช้คำสั่ง del
- ถ้าเกิน ไม่ดำเนินการใด ๆ

48. กำหนดให้ list4 = ['a', 'b', 'c', 'd', 'e', 'f'] เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระ (str1) จากนั้นให้ตรวจสอบว่า str1 เป็นสมาชิกของ list4 หรือไม่

- ถ้าเป็น ให้ทำการลบ str1 ออกจาก list4 โดยใช้เมธอด remove
- ถ้าไม่เป็น ไม่ดำเนินการใด ๆ

จากนั้นให้พิมพ์ค่า list4 ออกมา

49. กำหนดให้ list1 = [1, 2, 3, ... , 10] เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็มและให้ตรวจสอบว่า จำนวนที่รับมาเป็นสมาชิกของ list1 หรือไม่

- ถ้าเป็น ให้ทำการลบจำนวนนั้นโดยใช้คำสั่ง del
- ถ้าไม่เป็น ให้เพิ่มจำนวนนั้นโดยใช้เมธอด append

จากนั้นให้พิมพ์ค่า list1 ออกมา

12

50. กำหนดให้ list1 = [1, 2, 3, ... , 10] เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็มและให้ตรวจสอบว่า จำนวนที่รับมาเป็นสมาชิกของ list1 หรือไม่
    - ถ้าเป็น ให้ทำการลบจำนวนนั้นโดยใช้เมธอด remove
    - ถ้าไม่เป็น ให้เพิ่มจำนวนนั้นโดยใช้เมธอด append

    จากนั้นให้พิมพ์ค่า list1 ออกมา

51. กำหนดให้ list1 = [1, 2, 3, ... , 10] เขียนโปรแกรมลบสมาชิกที่ index เป็นเลขคู่โดยใช้คำสั่ง del จากนั้นให้พิมพ์ค่า list1 ออกมา

52. กำหนดให้ list1 = [1, 2, 3, ... , 10] เขียนโปรแกรมลบสมาชิกที่ index เป็นเลขคู่โดยใช้เมธอด remove จากนั้นให้พิมพ์ค่า list1 ออกมา

53. กำหนดให้ list3 = [1, 2, 3, ... , 25] เขียนโปรแกรมลบสมาชิกใน list3 ที่ค่าถูกหารด้วย 2 หรือ 3 ลงตัวเพียงตัวใดตัวหนึ่งโดยใช้คำสั่ง del จากนั้นให้พิมพ์ค่า list3 ที่ผ่านกระบวนการดังกล่าวออกมา

54. กำหนดให้ list3 = [1, 2, 3, ... , 25] เขียนโปรแกรมลบสมาชิกใน list3 ที่ค่าถูกหารด้วย 2 ลงตัวแต่หารด้วย 3 ไม่ลงตัวโดยใช้เมธอด remove จากนั้นให้พิมพ์ค่า list3 ที่ผ่านกระบวนการดังกล่าวออกมา

55. กำหนดให้ list1 = [1, 2, 3, ... , 10] เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม และตรวจสอบว่าจำนวนที่รับมาเป็นสมาชิกใน list1 หรือไม่
    - ถ้าเป็น ให้ทำการลบสมาชิกโดยใช้เมธอด remove
    - ถ้าไม่เป็น ไม่ดำเนินการใด ๆ

    กระบวนการนี้จะดำเนินไปเรื่อย ๆ จนกว่าความยาวของ list1 จะเท่ากับ 7 จากนั้นให้พิมพ์ค่า list1 ออกมา

56. กำหนดให้ matrix1 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] และ matrix2 = [[1, 2, 3], [1, 2, 3], [1, 2, 3]] เขียนโปรแกรมหาผลบวกของสมาชิกในตำแหน่งที่ตรงกันของ matrix1 และ matrix2 จากนั้นให้พิมพ์ผลลัพธ์ในรูปแบบของ list 2 มิติออกมา

12

57. กำหนดให้ matrix1 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] และ matrix2 = [[1, 2, 3], [1, 2, 3], [1, 2, 3]] เขียนโปรแกรมหาผลคูณของสมาชิกในตำแหน่งที่ตรงกันของ matrix1 และ matrix2 จากนั้นให้พิมพ์ผลลัพธ์ในรูปแบบของ list 2 มิติออกมา

58. กำหนดให้ matrix1 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] เขียนโปรแกรม transpose matrix1 และพิมพ์ผลลัพธ์ออกมา

59. กำหนดให้ matrix1 = [[1, 2, 3, 4], [5, 6, 7, 8]] เขียนโปรแกรม transpose matrix1 และพิมพ์ผลลัพธ์ออกมา

60. กำหนดให้ list1 = ['a', 'b', 'c', 'd', 'e', 'f'] เขียนโปรแกรมสร้าง tuple ที่มีสมาชิกเหมือนกับ list1 จากนั้นให้พิมพ์ tuple ที่สร้างออกมา

61. กำหนดให้ tuple1 = (1, 2, 3, ... , 10) เขียนโปรแกรมหาผลบวกกำลังสองของสมาชิกใน tuple1 ที่มี index เป็นเลขคู่ จากนั้นให้พิมพ์ผลลัพธ์ออกมา

62. กำหนดให้ tuple1 = (1, 2, 3, ... , 10) เขียนโปรแกรมหาผลบวกกำลังสองของสมาชิกใน tuple1 ที่มี index เป็นเลขคี่ จากนั้นให้พิมพ์ผลลัพธ์ออกมา

63. กำหนดให้ tuple2 = (1, 2, 3, ... , 25) เขียนโปรแกรมหาจำนวนเฉพาะใน tuple2 และพิมพ์ออกมา

64. กำหนดให้ tuple2 = (1, 2, 3, ... , 25) เขียนโปรแกรมหาค่าเฉลี่ยของจำนวนเฉพาะใน tuple2 และพิมพ์ออกมา

65. กำหนดให้ tuple1 = (1, 2, 3, ... , 10) เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็มและให้ทำการตรวจสอบว่าจำนวนที่รับมาเป็นสมาชิกของ tuple1 หรือไม่
    - ถ้าเป็น ให้พิมพ์ 'yes'
    - ถ้าไม่เป็น ให้พิมพ์ 'no'

12

66. กำหนดให้ tuple1 = (1, 2, 3, ... , 10) เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็มและให้ทำการตรวจสอบว่าความยาวของ tuple1 มีค่ามากกว่า, น้อยกว่า หรือเท่ากับจำนวนที่รับมา

- ถ้ามากกว่า ให้พิมพ์ 'มากกว่า'
- ถ้าน้อยกว่า ให้พิมพ์ 'น้อยกว่า'
- ถ้าเท่ากับ ให้พิมพ์ 'เท่ากับ'

67. กำหนดให้ list1 = ['one', 'two', 'three', 'four'] และ list2 = [1, 2, 3, 4] เขียนโปรแกรมสร้าง dictionary ที่มี key เป็นสมาชิกใน list1 และมี value เป็นสมาชิกใน list2 ที่ index ตรงกัน จากนั้นให้พิมพ์ค่า dictionary ที่สร้างออกมา

68. กำหนดให้ tuple1 = (1, 2, 3, 4) และ tuple2 = ('ant', 'bird', 'cat', 'dog') เขียนโปรแกรมสร้าง dictionary ที่มี key เป็นสมาชิกใน tuple1 และมี value เป็นสมาชิกใน tuple2 ที่ index ตรงกัน จากนั้นให้พิมพ์ค่า dictionary ที่สร้างออกมา

69. สร้าง empty dictionary และกำหนด key เป็น 0, 1, 2, ..., 9 โดยที่ value มีค่าเป็น 2 เท่าของ key จากนั้นให้พิมพ์ค่า dictionary ที่สร้างออกมา

70. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม (n) จากนั้นให้ผู้ใช้งานกรอกข้อมูลเพื่อสร้าง list1 และ list2 ความยาว n ที่สมาชิกทุกตัวเป็นสายอักขระและจำนวนเต็มตามลำดับ สุดท้ายให้สร้าง dictionary ที่มี key เป็นสมาชิกใน list1 และ value เป็นสมาชิกใน list2 ใน index ที่ตรงกันและพิมพ์ค่า dictionary ที่สร้างออกมา

12

71. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม (n) จากนั้นให้ผู้ใช้งานกรอกข้อมูลเพื่อสร้าง list1 และ list2 ความยาว n ที่สมาชิกทุกตัวเป็นสายอักขระและจำนวนเต็มตามลำดับ สุดท้ายให้สร้าง dictionary โดยมีเงื่อนไขดังนี้

- ถ้า index เป็นเลขคู่ ให้ key เป็นสมาชิกใน list1 และ value เป็นสมาชิกใน list2
- ถ้า index เป็นเลขคี่ ให้ key เป็นสมาชิกใน list2 และ value เป็นสมาชิกใน list1

72. เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นจำนวนเต็ม (n) จากนั้นให้ผู้ใช้งานกรอกข้อมูลเพื่อสร้าง list1 ความยาว n ที่มีสมาชิกเป็นจำนวนจริง จากนั้นหาค่าเฉลี่ยและความแปรปรวนของข้อมูลใน list1 และเก็บไว้ใน dictionary ที่มี key ชื่อ 'mean' และ 'variance' จากนั้นให้พิมพ์ค่า dictionary ที่สร้างออกมา

73. กำหนดให้ dict1 = {'apple' : '52 kcal', 'banana' : '132 kcal', 'carrot' : '46 kcal'} เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระและให้ตรวจสอบว่าสายอักขระที่รับมาเป็นหนึ่งใน key ของ dict1 หรือไม่
   - ถ้าเป็น ให้พิมพ์ value ของ key นั้นออกมา
   - ถ้าไม่เป็น ให้สร้าง key ใหม่ที่มีชื่อเดียวกับสายอักขระที่รับมา และกำหนด value เป็น 'wait for assignment' จากนั้นให้พิมพ์ค่า dict1 ออกมา

74. กำหนดให้ dict1 = {'apple' : '52 kcal', 'banana' : '132 kcal', 'carrot' : '46 kcal'} เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระและให้ทำการตรวจสอบว่า สายอักขระที่รับมาเป็นหนึ่งใน key ของ dict1 หรือไม่
   - ถ้าเป็น ให้แทนที่ value ของ key นั้นเป็น 'wait for new value'
   - ถ้าไม่เป็น ให้สร้าง key ใหม่ที่มีชื่อเดียวกับสายอักขระที่รับมา และกำหนด value เป็น 'wait for assignment'

   จากนั้นให้พิมพ์ค่า dict1 ออกมา

75. กำหนดให้ dict2 = {'name' : 'Mario', 'age' : '30', 'job' : 'mushroom picker'} เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระและทำการตรวจสอบว่า สายอักขระที่รับมาเป็นหนึ่งใน value ของ dict2 หรือไม่
   - ถ้าเป็น ให้พิมพ์ 'yes'
   - ถ้าไม่เป็น ให้พิมพ์ 'no'

76. กำหนดให้ dict2 = {'name' : 'Mario', 'age' : '30', 'job' : 'mushroom picker'} เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระและทำการตรวจสอบว่า สายอักขระที่รับมาเป็นหนึ่งใน value ของ dict2 หรือไม่
   - ถ้าเป็น ให้พิมพ์ key ของ value นั้นออกมา
   - ถ้าไม่เป็น ไม่ดำเนินการใด ๆ

12

77. กำหนดให้ dict2 = {'name' : 'Mario', 'age' : '30', 'job' : 'mushroom picker'} เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระและทำการตรวจสอบว่า สายอักขระที่รับมาเป็นหนึ่งใน value ของ dict2 หรือไม่
    - ถ้าเป็น ให้ลบ key ของ value นั้น
    - ถ้าไม่เป็น ให้สร้าง key ชื่อ 'new_key' และกำหนด value เป็นสายอักขระที่รับ

78. กำหนดให้ dict2 = {'name' : 'Mario', 'age' : '30', 'job' : 'mushroom picker'} และ list1 = ['first_name', 'last_name', 'age', 'job'] เขียนโปรแกรมลบ key ของ dict2 ที่เป็นสมาชิกของ list1 จากนั้นให้พิมพ์ค่า dict2 ออกมา

79. สร้าง empty dictionary (dict1) แล้วเขียนโปรแกรมรับอินพุตที่เป็นสายอักขระ จากนั้นให้ตรวจสอบว่าสายอักขระที่รับเป็นหนึ่งใน key ของ dict1 หรือไม่
    - ถ้าเป็น ให้ลบ key นั้น
    - ถ้าไม่เป็น ให้สร้าง key ชื่อเดียวกับสายอักขระที่รับ จากนั้นรับอินพุตที่เป็นจำนวนเต็มเพื่อมาเป็น value

80. กำหนดให้ list1 = ['a', 'b', 'c', 'd', 'e', 'f'] เขียนโปรแกรมสร้าง set ที่มีสมาชิกเหมือนกับ list1 จากนั้นให้พิมพ์ค่า set ที่สร้างออกมา

81. สร้าง empty set แล้วเขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนเต็มโดยใช้ for loop จากนั้นให้ทำการ add จำนวนที่รับลงใน set สุดท้ายให้พิมพ์ค่า set ที่สร้างและจำนวนสมาชิกใน set ออกมา

82. สร้าง empty set แล้วเขียนโปรแกรมรับอินพุตที่เป็นจำนวนเต็ม จากนั้นให้ทำการ add จำนวนที่รับลงใน set จนกว่า set จะมีสมาชิก 5 ตัว จากนั้นให้พิมพ์ค่า set ที่สร้างออกมา

12

83. ให้ set1 = {1, 2, 3, 'a', 'b', 'c'} เขียนโปรแกรมรับอินพุต 1 ตัวที่เป็นสายอักขระและให้ทำการตรวจสอบว่า สายอักขระที่รับมาเป็นหนึ่งในสมาชิกของ set1 หรือไม่
    - ถ้าเป็น ให้ทำการ remove สมาชิกตัวนั้น
    - ถ้าไม่เป็น ไม่ดำเนินการใด ๆ

    จากนั้นให้พิมพ์ค่า set1 ออกมา

84. กำหนดให้ list1 = ['t', 'u', 'r', 'i', 'n', 'g'] และ tuple1 = ('n', 'e', 'w', 't', 'o', 'n') เขียนโปรแกรมสร้าง set1 และ set2 โดยที่ set1 มีสมาชิกเป็น index เลขคู่ของ list1 และ set2 มีสมาชิกเป็น index เลขคี่ของ tuple1 จากนั้นให้พิมพ์ค่า set1 และ set2 ออกมา

85. จากผลลัพธ์ข้อ 84 ให้นำ set1 และ set2 มา union กัน จากนั้นให้พิมพ์ผลลัพธ์ออกมา

86. จากผลลัพธ์ข้อ 84 ให้นำ set1 และ set2 มา intersection กัน จากนั้นให้พิมพ์ผลลัพธ์ออกมา กัน

87. เขียนโปรแกรมรับอินพุตที่เป็นสายอักขระเพื่อสร้าง set1 และ set2 ที่มีสมาชิก 3 ตัว จากนั้นให้หา set1 - set2 และ set2 - set1 สุดท้ายให้พิมพ์ผลลัพธ์ออกมา

88. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนจริงโดยใช้ for loop จากนั้นให้หาค่าที่มากที่สุดและพิมพ์ออกมา

89. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนจริงโดยใช้ for loop จากนั้นให้หาค่าที่มากที่สุด, จำนวนครั้งที่ค่ามากสุดปรากฏ และพิมพ์ออกมา

90. กำหนดให้ country = ['Brazil', 'China', 'Germany', 'Japan', 'Sweden'] เขียนโปรแกรมเพื่อเก็บยอดขายสินค้า (จำนวนจริง) ลงใน dictionary โดยกำหนดให้ key เป็นสมาชิกใน country และ value เป็นยอดขายสินค้าของประเทศนั้น ๆ หลังจากทำการเก็บข้อมูลเรียบร้อย ให้พิมพ์ยอดขายสูงสุดและประเทศที่ทำยอดขายได้สูงสุดออกมา

91. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนจริงโดยใช้ for loop จากนั้นให้หาค่าที่น้อยที่สุดและพิมพ์ออกมา

12

92. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนจริงโดยใช้ for loop จากนั้นให้หาค่าที่น้อยที่สุด, จำนวนครั้งที่ค่าน้อยสุดปรากฏ และพิมพ์ออกมา

93. กำหนดให้ country = ['Brazil', 'China', 'Germany', 'Japan', 'Sweden'] เขียนโปรแกรมเพื่อเก็บยอดขายสินค้า (จำนวนจริง) ลงใน dictionary โดยกำหนดให้ key เป็นสมาชิกใน country และ value เป็นยอดขายสินค้าของประเทศนั้น ๆ หลังจากทำการเก็บข้อมูลเรียบร้อย ให้พิมพ์ยอดขายต่ำสุดและประเทศที่ทำยอดขายได้ต่ำสุดออกมา

94. กำหนดให้ list1 = [1, 3, 7, 9, 2, 7, 5, 1, 8] เขียนโปรแกรมหามัธยฐานของข้อมูลใน list1

95. กำหนดให้ list1 = [6, 5, 7, 9, 8, 1, 3, 5, 7, 2] เขียนโปรแกรมหามัธยฐานของข้อมูลใน list1

96. กำหนดให้ data = {'Brazil' : 35000, 'China' : 24000, 'Germany' : 42000, 'Japan' : 53000, 'Sweden' : 17000} เขียนโปรแกรมหามัธยฐานของข้อมูลใน data

97. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนจริงโดยใช้ for loop จากนั้นให้พิมพ์อินพุตที่แตกต่างกันทั้งหมดออกมา

12

98. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนจริงโดยใช้ for loop จากนั้นให้สร้าง dictionary ที่มี key เป็นจำนวนจริงที่รับมา (ที่แตกต่างกัน) และ value เป็นจำนวนครั้งที่จำนวนนั้น ๆ ปรากฏ สุดท้ายให้พิมพ์ค่า dictionary ที่สร้างออกมา

99. เขียนโปรแกรมรับอินพุต 5 ตัวที่เป็นจำนวนจริงโดยใช้ for loop จากนั้นให้ทำการหาฐานนิยมและพิมพ์ออกมา

100. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
xxxxxxxxxx
```

101. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
```

102. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
x
xx
xxx
xxxx
xxxxx
xxxxxx
xxxxxxx
xxxxxxxx
xxxxxxxxx
xxxxxxxxxx
```

103. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
xxxxxxxxxx
xxxxxxxxx
xxxxxxxx
xxxxxxx
xxxxxx
xxxxx
xxxx
xxx
xx
x
```

12

104. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
---x
--xxx
-xxxxx
xxxxxxx
```

105. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
xxxxxxx
-xxxxx
--xxx
---x
```

106. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
x-----x
-x---x-
--x-x--
---x---
--x-x--
-x---x-
x-----x
```

107. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for–while loop

```
---x---
--x-x--
-x---x-
x-----x
-x---x-
--x-x--
---x---
```

108. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for–while loop

```
---------x
--------x
-------x
------x
-----x
----x
---x
--x
-x
x
```

109. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
---------x
--------xx
-------xxx
------xxxx
-----xxxxx
----xxxxxx
---xxxxxxx
--xxxxxxxx
-xxxxxxxxx
xxxxxxxxxx
```

110. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
---------x
--------xxx
-------xxxxx
------xxxxxxx
-----xxxxxxxxx
----xxxxxxxxxxx
---xxxxxxxxxxxxx
--xxxxxxxxxxxxxxx
-xxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxx
```

12

111. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
---------x----------
--------xxx---------
-------xxxxx--------
------xxxxxxx-------
-----xxxxxxxxx------
----xxxxxxxxxxx-----
---xxxxxxxxxxxxx----
--xxxxxxxxxxxxxxx---
-xxxxxxxxxxxxxxxxx--
xxxxxxxxxxxxxxxxxxx-
```

112. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for–while loop

```
--------1
-------12
------123
-----1234
----12345
---123456
--1234567
-12345678
123456789
```

113. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for–while loop

```
--------1
-------121
------12321
-----1234321
----123454321
---12345654321
--1234567654321
-123456787654321
12345678987654321
```

114. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

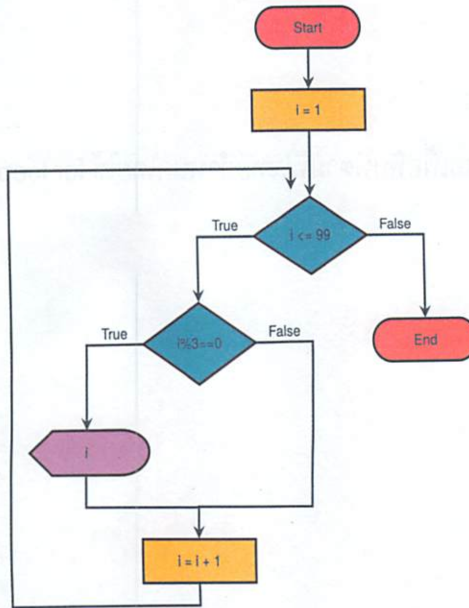
```
----1
---123
--54321
-1234567
987654321
```

115. เขียนโปรแกรมเพื่อพิมพ์ตามที่โจทย์กำหนดโดยใช้ for loop

```
----1
---12
--543
-1234
98765
```

12.2 Solution

เฉลยข้อที่ 1.



Input :

```
1 for i in range(1, 100):
2     if i%3 == 0:
3         print(i)
```

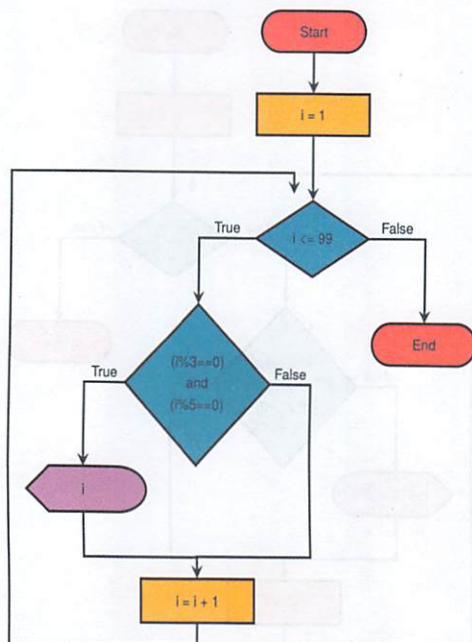
Output :

```
3
6
9
:
99
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 1 ถึง 99
บรรทัดที่ 2-3 : ถ้า i หารด้วย 3 ลงตัว ให้พิมพ์ค่า i ออกมา

เฉลยข้อที่ 2.



Input :

```
1 for i in range(1, 100):
2     if (i%3 == 0) and (i%5 == 0):
3         print(i)
```

Output :

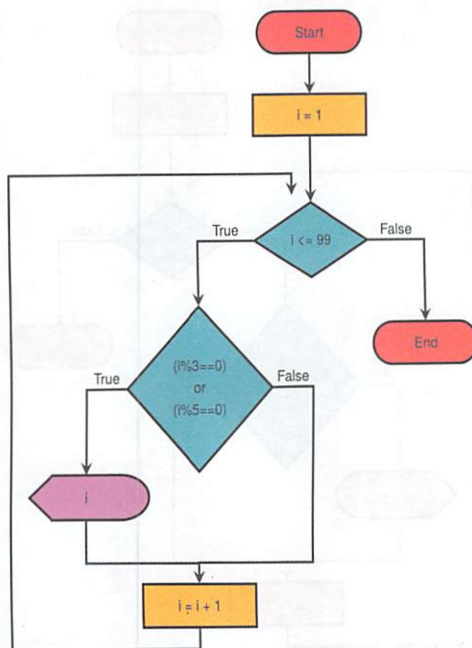
```
15
30
45
:
90
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 1 ถึง 99
บรรทัดที่ 2-3 : ถ้า i หารด้วย 3 และ 5 ลงตัว ให้พิมพ์ค่า i ออกมา

12

เฉลยข้อที่ 3.



Input :

```
1 for i in range(1, 100):
2     if (i%3 == 0) or (i%5 == 0):
3         print(i)
```

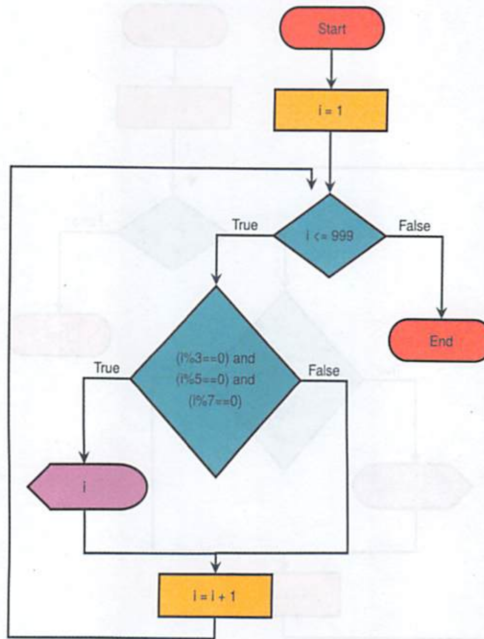
Output :

```
3
5
6
:
99
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 1 ถึง 99
บรรทัดที่ 2-3 : ถ้า i หารด้วย 3 หรือ 5 ลงตัว ให้พิมพ์ค่า i ออกมา

เฉลยข้อที่ 4.



Input :

```
1 for i in range(1, 1000):
2     if (i%3 == 0) and (i%5 == 0) and (i%7 == 0):
3         print(i)
```

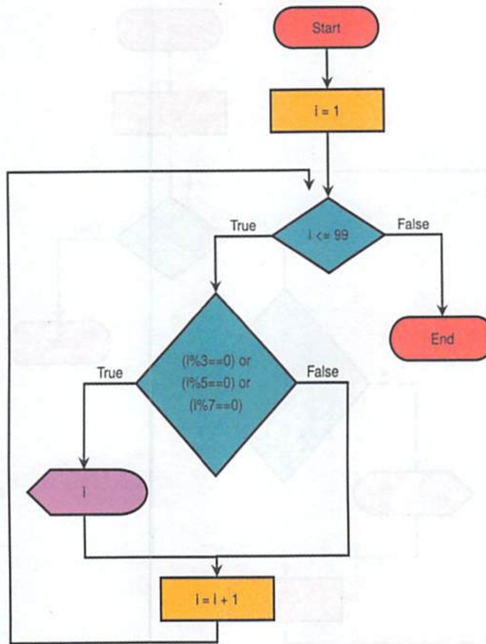
Output :

```
105
210
315
:
945
```


เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 1 ถึง 999
บรรทัดที่ 2-3 : ถ้า i หารด้วย 3, 5 และ 7 ลงตัว ให้พิมพ์ค่า i ออกมา

เฉลยข้อที่ 5.



Input :

```
1 for i in range(1, 100):
2     if (i%3 == 0) or (i%5 == 0) or (i%7 == 0):
3         print(i)
```

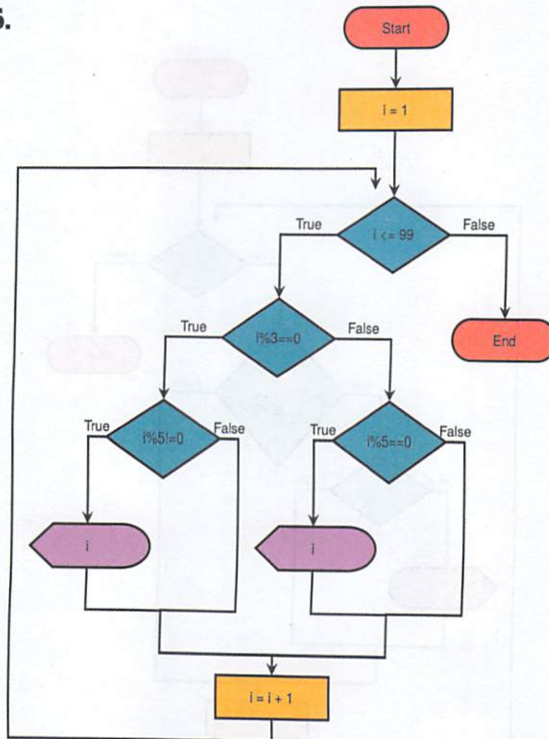

Output :

```
3
5
6
7
:
99
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 1 ถึง 99
บรรทัดที่ 2-3 : ถ้า i หารด้วย 3, 5 หรือ 7 ลงตัว ให้พิมพ์ค่า i ออกมา

เฉลยข้อที่ 6.



Input :

```
1 for i in range(1, 100):
2     if i%3 == 0:
3         if i%5 != 0:
4             print(i)
5     else:
6         if i%5 == 0:
7             print(i)
```

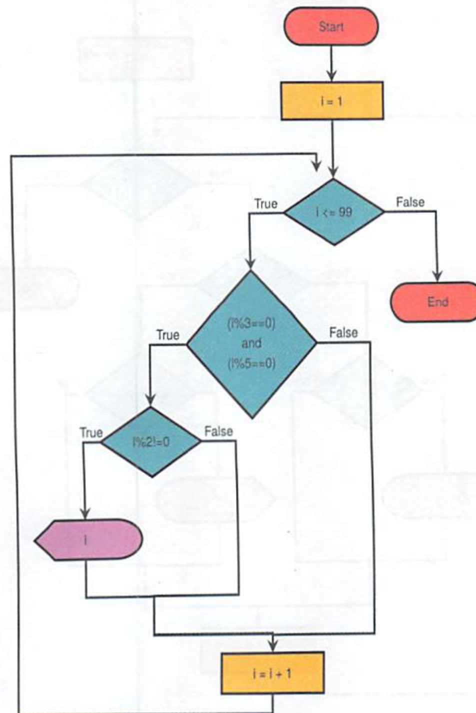
Output :

```
3
5
:
99
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 1 ถึง 99
บรรทัดที่ 2 : ถ้า i หารด้วย 3 ลงตัว ให้ทำบรรทัดที่ 3-4
บรรทัดที่ 3-4 : ถ้า i หารด้วย 5 ไม่ลงตัว ให้พิมพ์ค่า i ออกมา
บรรทัดที่ 5 : ถ้า i หารด้วย 3 ไม่ลงตัว ให้ทำบรรทัดที่ 6-7
บรรทัดที่ 6-7 : ถ้า i หารด้วย 5 ลงตัว ให้พิมพ์ค่า i ออกมา

เฉลยข้อที่ 7.



Input :

```
1 for i in range(1, 100):
2     if (i%3 == 0) and (i%5 == 0):
3         if i%2 != 0:
4             print(i)
```


Output :

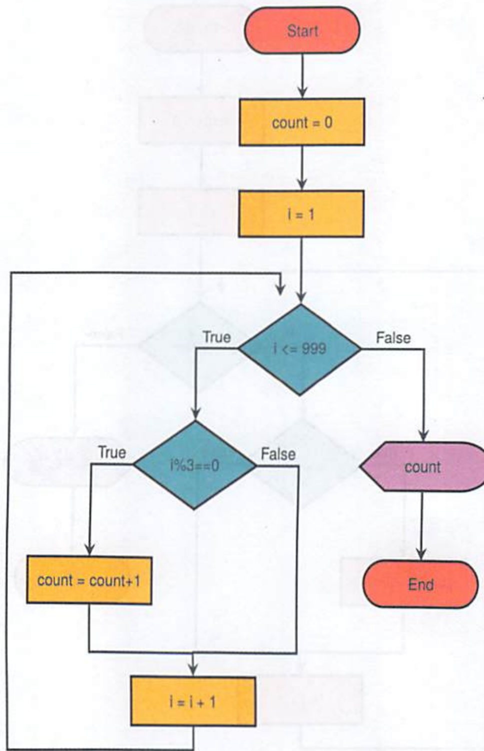
```
15
45
75
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 1 ถึง 99
บรรทัดที่ 2 : ถ้า i หารด้วย 3 และ 5 ลงตัว ให้ทำบรรทัดที่ 3-4
บรรทัดที่ 3-4 : ถ้า i หารด้วย 2 ไม่ลงตัว ให้พิมพ์ค่า i ออกมา

เฉลยข้อที่ 8. แนวคิดเดียวกับข้อ 7

เฉลยข้อที่ 9.



Input :

```
1 count = 0
2 for i in range(1, 1000):
3     if i%3 == 0:
4         count = count + 1
5 print(count)
```

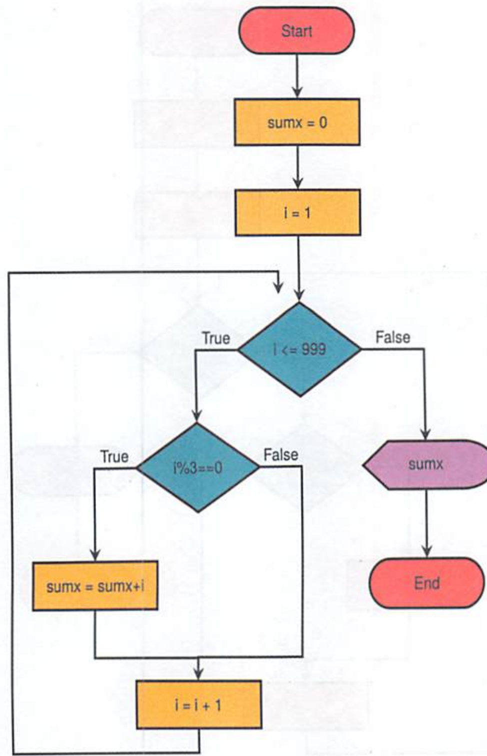
Output :

```
333
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ count และกำหนดค่าเท่ากับ 0
บรรทัดที่ 2 : พิจารณาค่า i ตั้งแต่ 1 ถึง 999
บรรทัดที่ 3-4 : ถ้า i หารด้วย 3 ลงตัว ให้เพิ่ม count ขึ้น 1
บรรทัดที่ 5 : พิมพ์ค่า count ออกมา

เฉลยข้อที่ 10.



12

Input :

```
1 sumx = 0
2 for i in range(1, 1000):
3     if i%3 == 0:
4         sumx = sumx + i
5 print(sumx)
```

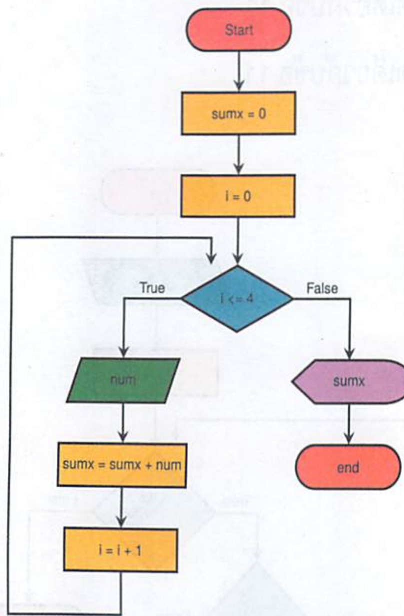
Output :

```
166833
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ sumx และกำหนดค่าเท่ากับ 0
บรรทัดที่ 2 : พิจารณาค่า i ตั้งแต่ 1 ถึง 999
บรรทัดที่ 3-4 : ถ้า i หารด้วย 3 ลงตัว ให้นำ i ไปบวกเพิ่มที่ sumx
บรรทัดที่ 5 : พิมพ์ค่า sumx ออกมา

เฉลยข้อที่ 11.



Input :

```
1 sumx = 0
2 for i in range(5):
3     num = int(input('Please insert num' + str(i+1) + ': '))
4     sumx = sumx + num
5 print(sumx)
```

Output :

```
Please insert num1: 1
Please insert num2: 3
Please insert num3: 5
Please insert num4: 7
Please insert num5: 9
25
```

เฉลยละเอียด

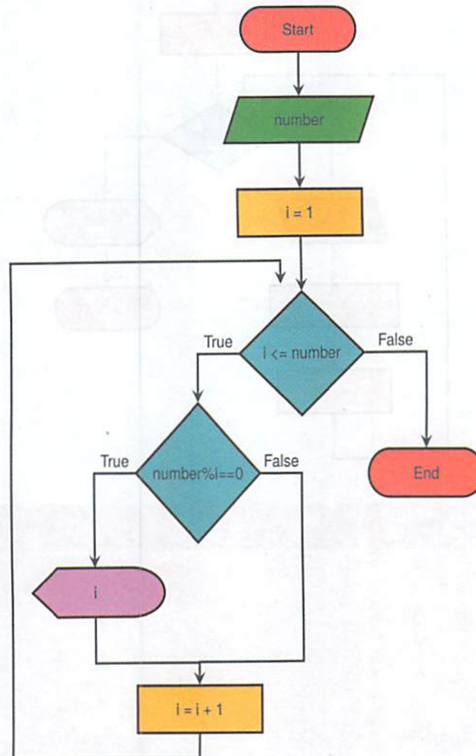
บรรทัดที่ 1 : สร้างตัวแปรชื่อ sumx และกำหนดค่าเท่ากับ 0
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-4 ซ้ำ 5 ครั้ง
บรรทัดที่ 3-4 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ num จากนั้นนำไปบวกเพิ่มที่ sumx
บรรทัดที่ 5 : พิมพ์ค่า sumx ออกมา

12

เฉลยข้อที่ 12. แนวคิดเดียวกับข้อ 11

เฉลยข้อที่ 13. แนวคิดเดียวกับข้อ 11

เฉลยข้อที่ 14.



Input :

```
1 number = int(input('Please insert number: '))
2 for i in range(1, number + 1):
3     if number%i == 0:
4         print(i)
```

Output :

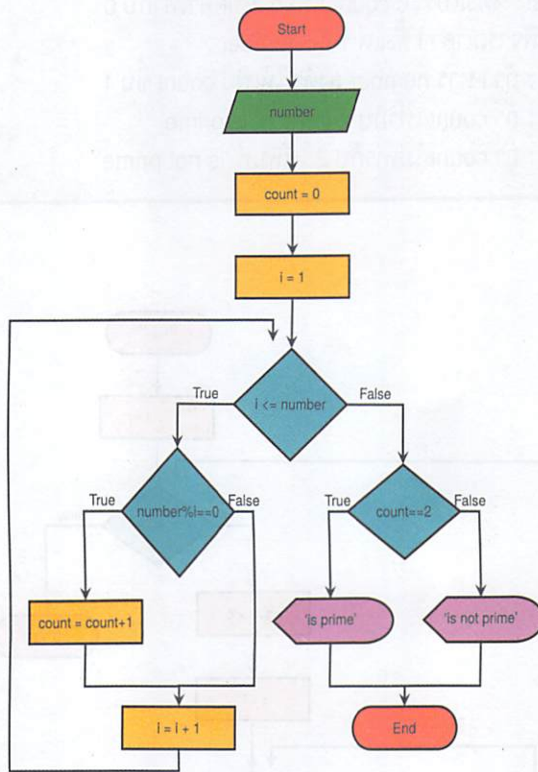
```
Please insert number: 135
1
3
45
135
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 2 : พิจารณาค่า i ตั้งแต่ 1 ถึง number
บรรทัดที่ 3-4 : ถ้า i หาร number ลงตัว ให้พิมพ์ i ออกมา

เฉลยข้อที่ 15. แนวคิดเดียวกับข้อ 9

เฉลยข้อที่ 16.



Input :

```
number = int(input('Please insert number: '))
count = 0
for i in range(1, number + 1):
    if number%i == 0:
        count = count + 1
if count == 2:
    print('is prime')
else:
    print('is not prime')
```

Output :

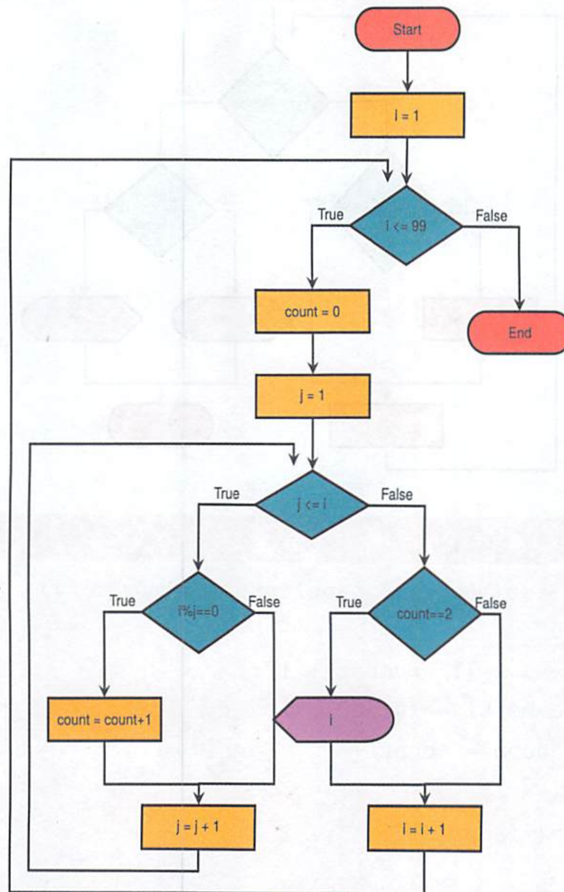
```
Please insert number: 135
is not prime
```

12

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 2 : สร้างตัวแปรชื่อ count และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3 : พิจารณาค่า i ตั้งแต่ 1 ถึง number
บรรทัดที่ 4-5 : ถ้า i หาร number ลงตัว ให้เพิ่ม count ขึ้น 1
บรรทัดที่ 6-7 : ถ้า count เท่ากับ 2 ให้พิมพ์ 'is prime'
บรรทัดที่ 8-9 : ถ้า count ไม่เท่ากับ 2 ให้พิมพ์ 'is not prime'

เฉลยข้อที่ 17.



Input :

```
for i in range(1, 100):
    count = 0
    for j in range(1, i + 1):
        if i%j == 0:
            count = count + 1
    if count == 2:
        print(i)
```

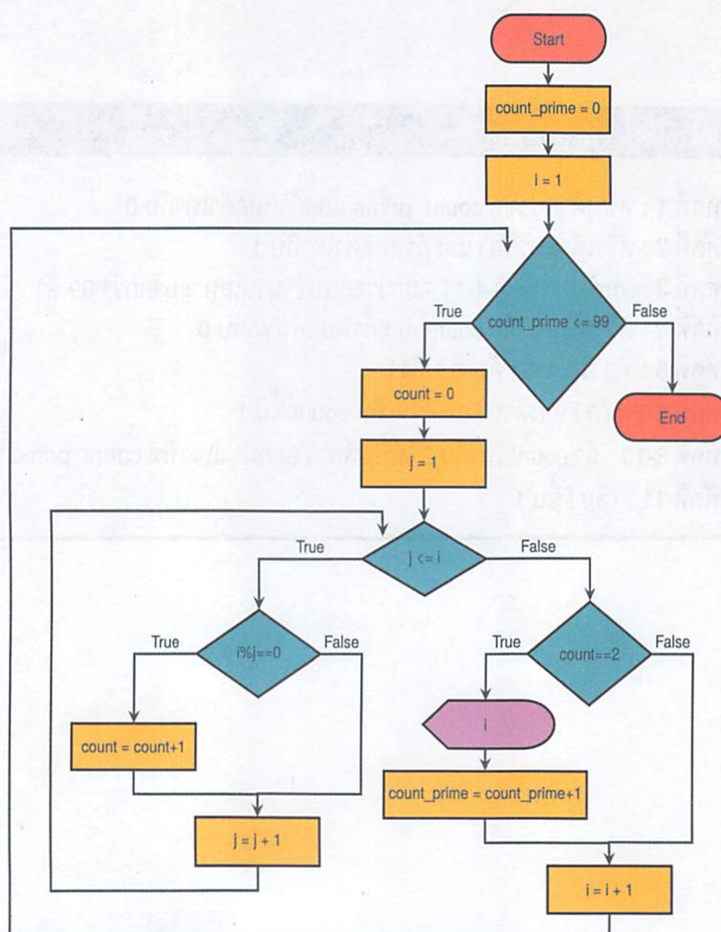
Output :

```
2
3
5
:
97
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 1 ถึง 99
บรรทัดที่ 2 : สร้างตัวแปรชื่อ count และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3 : พิจารณาค่า j ตั้งแต่ 1 ถึง i
บรรทัดที่ 4-5 : ถ้า i หารด้วย j ลงตัว ให้เพิ่ม count ขึ้น 1
บรรทัดที่ 6-7 : ถ้า count เท่ากับ 2 ให้พิมพ์ค่า i ออกมา

เฉลยข้อที่ 18.



12

Input :

```
1 count_prime = 0
2 i = 1
3 while count_prime <= 99:
4     count = 0
5     for j in range(1, i + 1):
6         if i%j == 0:
7             count = count + 1
8     if count == 2:
9         print(i)
10        count_prime = count_prime + 1
11    i = i + 1
```

Output :

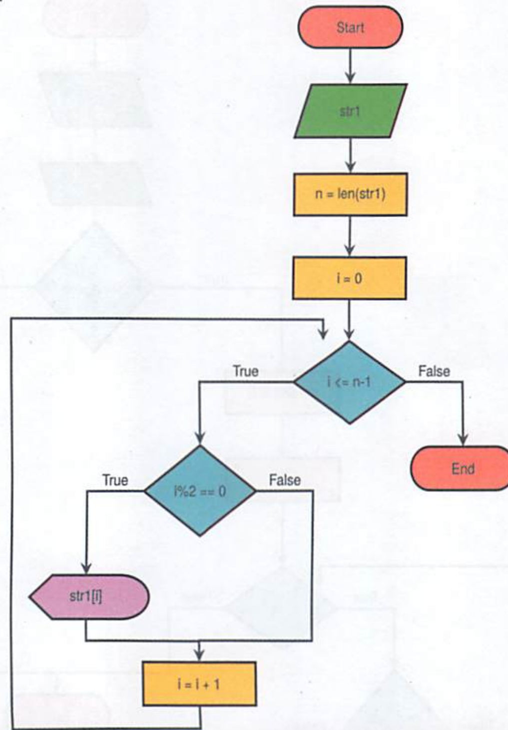
```
2
3
5
:
541
```


เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ count_prime และกำหนดค่าเท่ากับ 0
บรรทัดที่ 2 : สร้างตัวแปรชื่อ i และกำหนดค่าเท่ากับ 1
บรรทัดที่ 3 : วนทำบรรทัดที่ 4-11 จนกว่าจะพบจำนวนเฉพาะมากกว่า 99 ตัว
บรรทัดที่ 4 : สร้างตัวแปรชื่อ count และกำหนดค่าเท่ากับ 0
บรรทัดที่ 5 : พิจารณาค่า j ตั้งแต่ 1 ถึง i
บรรทัดที่ 6-7 : ถ้า i หารด้วย j ลงตัวให้เพิ่ม count ขึ้น 1
บรรทัดที่ 8-10 : ถ้า count เท่ากับ 2 ให้พิมพ์ค่า i ออกมา และเพิ่ม count_prime ขึ้น 1
บรรทัดที่ 11 : เพิ่ม i ขึ้น 1

เฉลยข้อที่ 19. แนวคิดเดียวกับข้อ 11 และ 18

เฉลยข้อที่ 20.



Input :

```
1 str1 = input('Please insert str1: ')
2 n = len(str1)
3 for i in range(n):
4     if i%2 == 0:
5         print(str1[i])
```

Output :

```
Please insert str1: coding
c
d
n
```

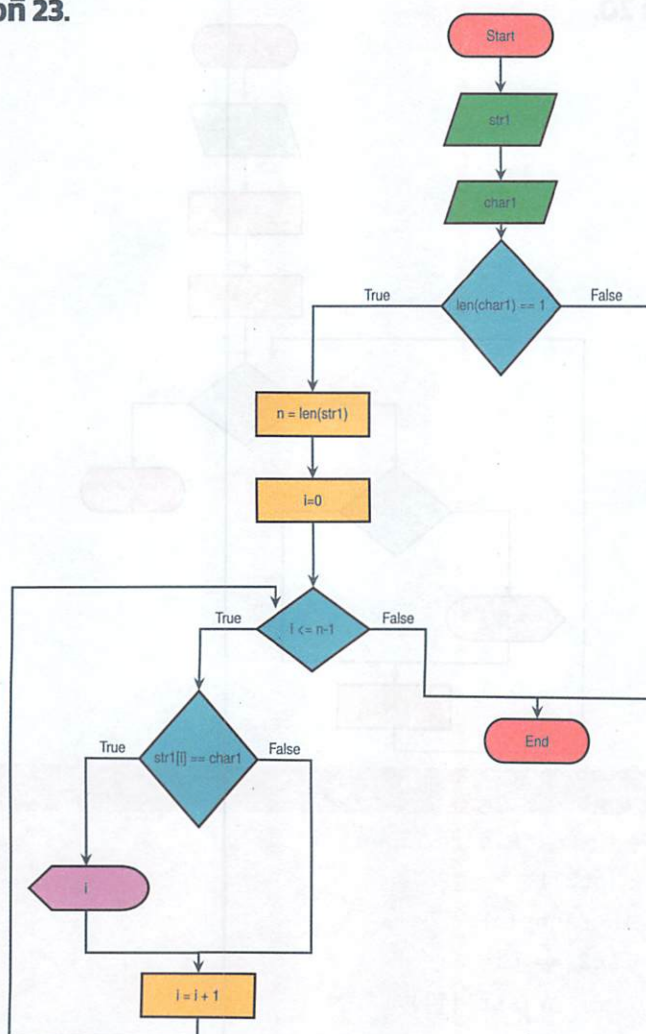
เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปร str1
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ str1
บรรทัดที่ 3 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 4-5 : ถ้า i หารด้วย 2 ลงตัว ให้พิมพ์ค่า str1 ใน index ที่ i ออกมา

เฉลยข้อที่ 21. แนวคิดเดียวกับข้อ 20

เฉลยข้อที่ 22. แนวคิดเดียวกับข้อ 11

เฉลยข้อที่ 23.



Input :

```
str1 = input('Please insert str1: ')
char1 = input('Please insert char: ')
if len(char1) == 1:
    n = len(str1)
    for i in range(n):
        if str1[i] == char1:
            print(i)
```

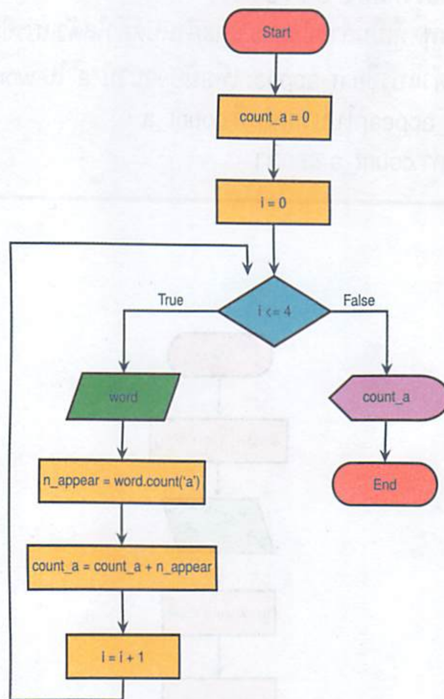
Output :

```
Please insert str1: google
Please insert char: o
1
2
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 2 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ char1
บรรทัดที่ 3 : ถ้าความยาวของ char1 เท่ากับ 1 ให้ทำบรรทัดที่ 4-7
บรรทัดที่ 4 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ char1
บรรทัดที่ 5-7 : พิจารณาอักขระแต่ละตัวใน str1 ถ้ามีค่าเท่ากับ char1 ให้พิมพ์ค่า i ออกมา

เฉลยข้อที่ 24.



Input :

```
count_a = 0
for i in range(5):
    word = input('Please insert word: ')
    n_appear = word.count('a')
    count_a = count_a + n_appear
print(count_a)
```

Output :

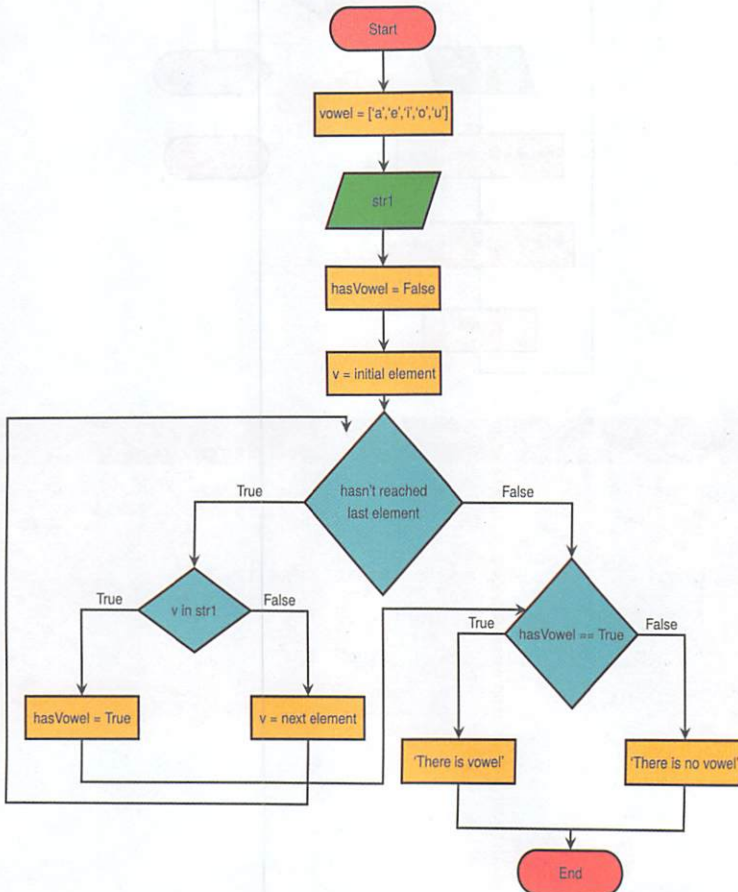
```
Please insert word: coconut
Please insert word: watermelon
Please insert word: lemon
Please insert word: mango
Please insert word: mangoteen
3
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ count_a และกำหนดค่าเท่ากับ 0
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-5 ซ้ำ 5 ครั้ง
บรรทัดที่ 3 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ word
บรรทัดที่ 4 : สร้างตัวแปรชื่อ n_appear เพื่อนับจำนวน 'a' ใน word
บรรทัดที่ 5 : นำ n_appear ไปบวกเพิ่มที่ count_a
บรรทัดที่ 6 : พิมพ์ค่า count_a ออกมา

12

เฉลยข้อที่ 25.



Input :

```
1 vowel = ['a', 'e', 'i', 'o', 'u']
2 str1 = input('Please insert str1: ')
3 hasVowel = False
4 for v in vowel:
5     if v in str1:
6         hasVowel = True
7         break
8 if hasVowel == True:
9     print('There is vowel')
10 else:
11     print('There is no vowel')
```

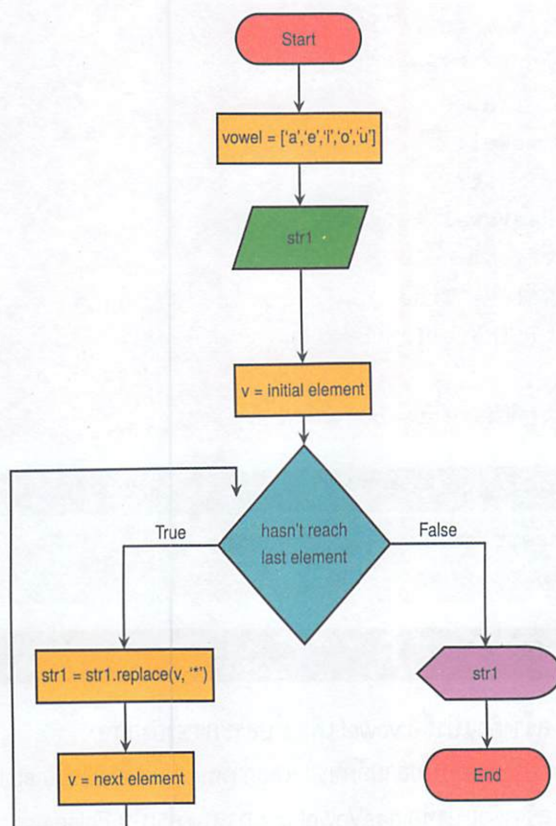
Output :

```
Please insert str1: I love coding
There is vowel
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ vowel เพื่อเก็บสระภาษาอังกฤษ
บรรทัดที่ 2 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 3 : สร้างตัวแปรชื่อ hasVowel และกำหนดค่าเป็น False
บรรทัดที่ 4 : พิจารณาสมาชิกแต่ละตัวใน vowel ในชื่อตัวแปร v
บรรทัดที่ 5-7 : ถ้า v ปรากฏใน str1 ให้ hasVowel เป็น True และทำการ break ออกจาก loop ทันที
บรรทัดที่ 8-9 : ถ้า hasVowel เป็น True ให้พิมพ์ 'There is vowel'
บรรทัดที่ 10-11 : ถ้า hasVowel เป็น False ให้พิมพ์ 'There is no vowel'

เฉลยข้อที่ 26.



Input :

```
1 vowel = ['a', 'e', 'i', 'o', 'u']
2 str1 = input('Please insert str1: ')
3 for v in vowel:
4     str1 = str1.replace(v, '*')
5 print(str1)
```

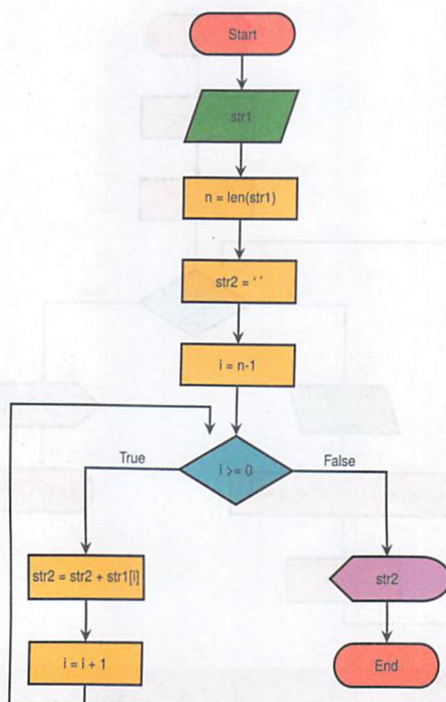
Output :

```
Please insert str1: I love coding
I l*v* c*d*ng
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ vowel เพื่อเก็บสระภาษาอังกฤษ
บรรทัดที่ 2 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 3 : พิจารณาสมาชิกแต่ละตัวใน vowel ในชื่อตัวแปร v
บรรทัดที่ 4 แทนที่ v ใน str1 ด้วย *
บรรทัดที่ 5 : พิมพ์ค่า str1 ออกมา

เฉลยข้อที่ 27.



Input :

```
1 str1 = input('Please insert str1: ')
2 n = len(str1)
3 str2 = ''
4 for i in range(n-1, -1, -1):
5     str2 = str2 + str1[i]
6 print(str2)
```

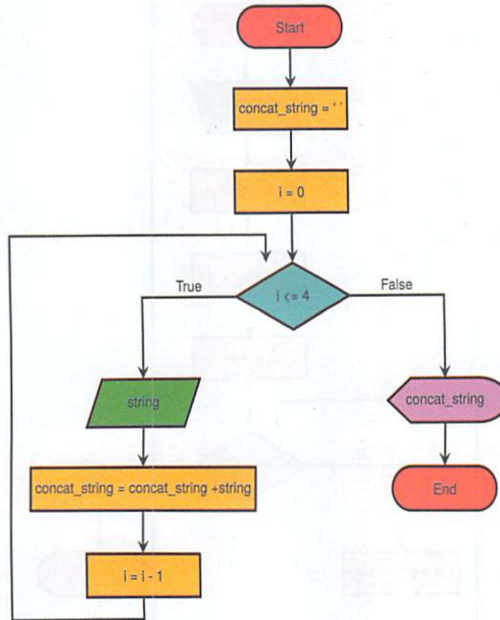
Output :

```
Please insert str1: I love coding
gnidoc evol I
```


เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ str1
บรรทัดที่ 3 : สร้างตัวแปรชื่อ str2 และกำหนดให้เป็น empty string
บรรทัดที่ 4-5 : พิจารณาอักขระแต่ละตัวใน str1 จากหลังมาหน้า แล้วนำอักขระไปต่อกับ str2
บรรทัดที่ 6 : พิมพ์ค่า str2 ออกมา

เฉลยข้อที่ 28.



Input :

```
1 concat_string = ''
2 for i in range(5):
3     string = input('Please insert string: ')
4     concat_string = concat_string + string
5 print(concat_string)
```

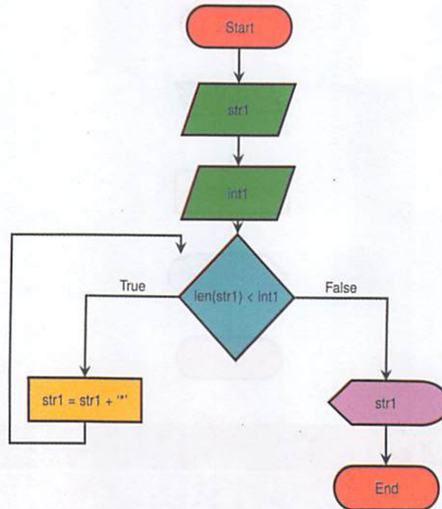

Output :

```
Please insert string: abc
Please insert string: defg
Please insert string: hij
Please insert string: klmnop
Please insert string: qrstuv
abcdefghijklmnopqrstuv
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ concat_string และกำหนดให้เป็น empty string
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-4 ซ้ำ 5 ครั้ง
บรรทัดที่ 3 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ string
บรรทัดที่ 4 : นำ string ที่รับมาไปต่อกับ concat_string
บรรทัดที่ 5 : พิมพ์ค่า concat_string ออกมา

เฉลยข้อที่ 29.



Input :

```
1 str1 = input('Please insert str1: ')
2 int1 = int(input('Please insert int1: '))
3 while len(str1) < int1:
4     str1 = str1 + '*'
5 print(str1)
```

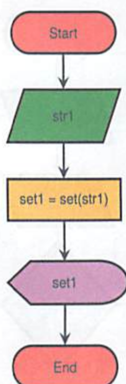
Output :

```
Please insert str1: abc
Please insert int1: 5
abc**
```


เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 2 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ int1
บรรทัดที่ 3 : วนทำบรรทัดที่ 4 จนกว่าความยาวของ str1 จะมากกว่าหรือเท่ากับ int1
บรรทัดที่ 4 : นำ '*' ไปต่อจาก str1 เดิม
บรรทัดที่ 5 : พิมพ์ค่า str1 ออกมา

เฉลยข้อที่ 30.



Input :

```
1 str1 = input('Please insert str1: ')
2 set1 = set(str1)
3 print(set1)
```

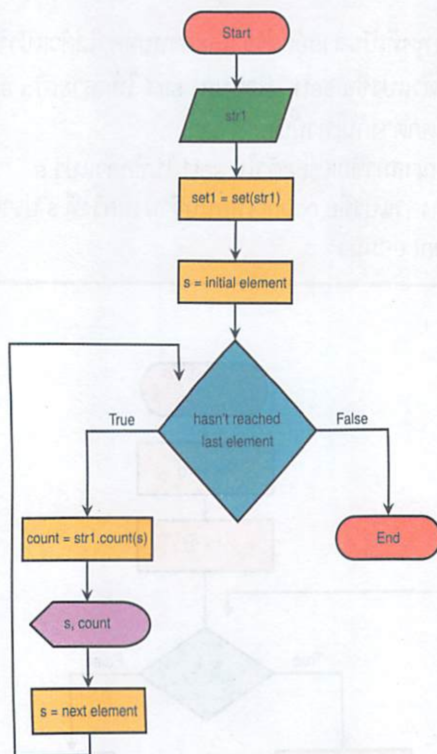
Output :

```
Please insert str1: abcabc
{'a', 'b', 'c'}
```

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 2 : สร้างตัวแปรชื่อ set1 เพื่อแปลง str1 ให้กลายเป็น set (ภายใน set จะเก็บเพียงสมาชิกที่แตกต่างกันเท่านั้น)
บรรทัดที่ 3 : พิมพ์ค่า set1 ออกมา

เฉลยข้อที่ 31.



Input :

```
str1 = input('Please insert str1: ')
set1 = set(str1)
for s in set1:
    count = str1.count(s)
    print(s, count)
```

Output :

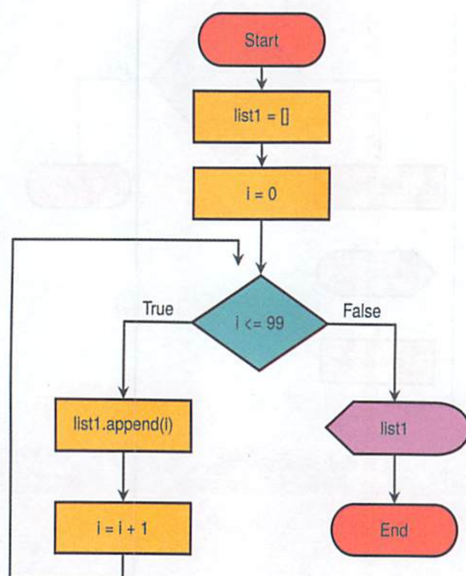
```
Please insert str1: alligator
a 2
r 1
g 1
i 1
l 2
o 1
t 1
```

12

เฉลยละเอียด

บรรทัดที่ 1 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 2 : สร้างตัวแปรชื่อ set1 เพื่อแปลง str1 ให้กลายเป็น set (ภายใน set จะเก็บเพียงสมาชิกที่แตกต่างกันเท่านั้น)
บรรทัดที่ 3 : พิจารณาสมาชิกแต่ละตัวใน set1 ในชื่อตัวแปร s
บรรทัดที่ 4-5 : สร้างตัวแปรชื่อ count เพื่อนับจำนวนครั้งที่ s ปรากฏขึ้นใน str1 แล้วพิมพ์ค่า s และ count ออกมา

เฉลยข้อที่ 32.



Input :

```
1 list1 = []
2 for i in range(20):
3     list1.append(i)
4 print(list1)
```

Output :

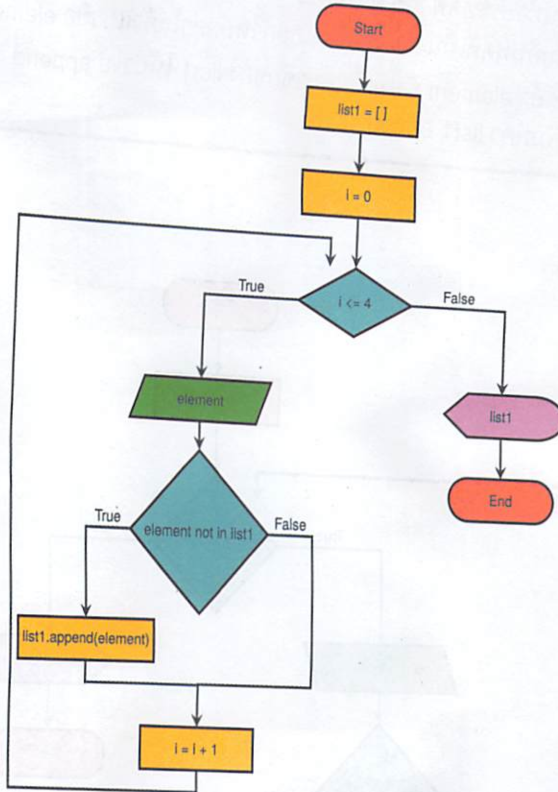
```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15,
16, 17, 18, 19]
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 และกำหนดให้เป็น empty list
บรรทัดที่ 2 : พิจารณาค่า i ตั้งแต่ 0 ถึง 19
บรรทัดที่ 3 : นำ i ไป append ใน list1
บรรทัดที่ 4 : พิมพ์ค่า list1 ออกมา

เฉลยข้อที่ 33. แนวคิดเดียวกับข้อ 32

เฉลยข้อที่ 34.



Input :

```
list1 = []
for i in range(5):
    element = int(input('Please insert element: '))
    if element not in list1:
        list1.append(element)
print(list1)
```

Output :

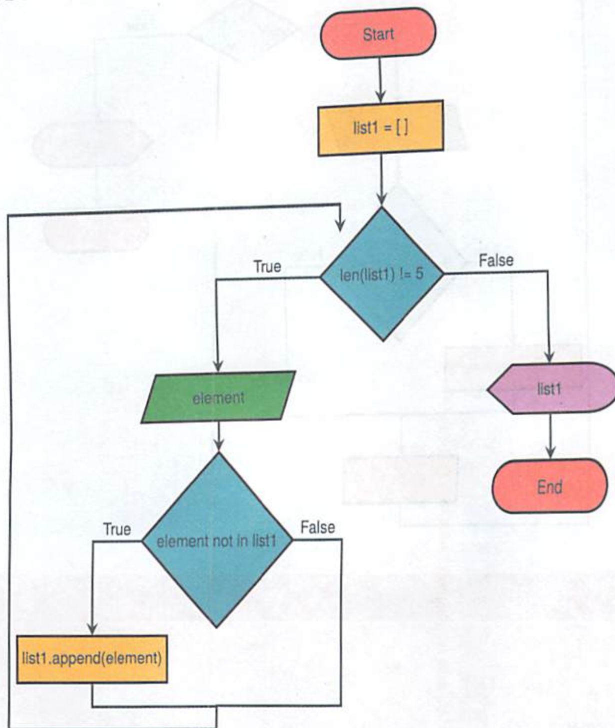
```
Please insert element: 5
Please insert element: 2
Please insert element: 3
Please insert element: 4
Please insert element: 2
[5, 2, 3, 4]
```

12

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 และกำหนดให้เป็น empty list
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-5 ซ้ำ 5 ครั้ง
บรรทัดที่ 3 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ element
บรรทัดที่ 4-5 : ถ้า element ยังไม่เป็นสมาชิกของ list1 ให้นำไป append ใน list1
บรรทัดที่ 6 : พิมพ์ค่า list1 ออกมา

เฉลยข้อที่ 35.



Input :

```
1 list1 = []
2 while len(list1) != 5:
3     element = input('Please insert element: ')
4     if element not in list1:
5         list1.append(element)
6 print(list1)
```

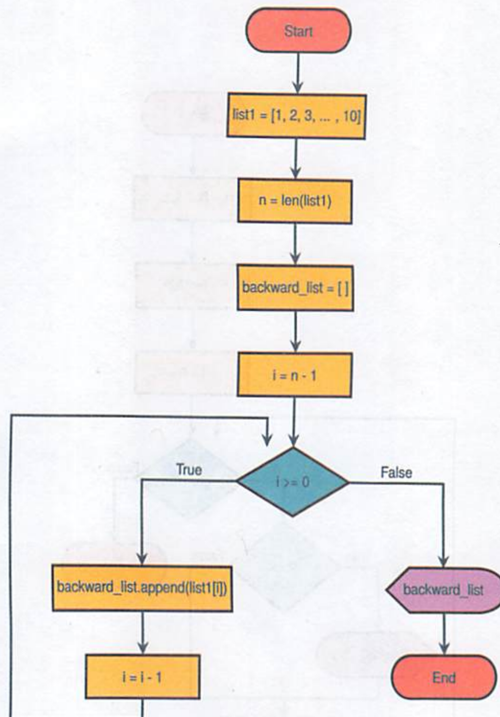
Output :

```
Please insert element: 1
Please insert element: 2
Please insert element: 2
Please insert element: 1
Please insert element: 12
Please insert element: 3
Please insert element: 5
['1', '2', '12', '3', '5']
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 และกำหนดให้เป็น empty list
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-5 ซ้ำจนกว่าสมาชิกใน list1 จะครบ 5 ตัว
บรรทัดที่ 3 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ element
บรรทัดที่ 4-5 : ถ้า element ยังไม่เป็นสมาชิกของ list1 ให้นำไป append ใน list1
บรรทัดที่ 6 : พิมพ์ค่า list1 ออกมา

เฉลยข้อที่ 36.



12

Input :

```
1 list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
2 n = len(list1)
3 backward_list = []
4 for i in range(n-1, -1, -1):
5     backward_list.append(list1[i])
6 print(backward_list)
```

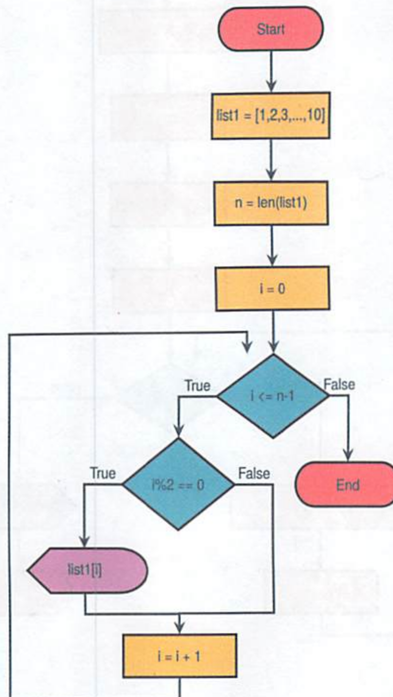
Output :

```
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 เพื่อเก็บเลข 1 ถึง 10
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list1
บรรทัดที่ 3 : สร้างตัวแปรชื่อ backward_list และกำหนดให้เป็น empty list
บรรทัดที่ 4 : พิจารณาค่า i ตั้งแต่ n-1 ถึง 0 แบบถอยหลัง
บรรทัดที่ 5 : นำสมาชิกของ list1 ในตำแหน่งที่ i ไป append ใน backward_list
บรรทัดที่ 6 : พิมพ์ค่า backward_list ออกมา

เฉลยข้อที่ 37.



Input :

```
1 list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
2 n = len(list1)
3 for i in range(n):
4     if i%2 == 0:
5         print(list1[i])
```

Output :

```
1
3
5
7
9
```

เฉลยละเอียด

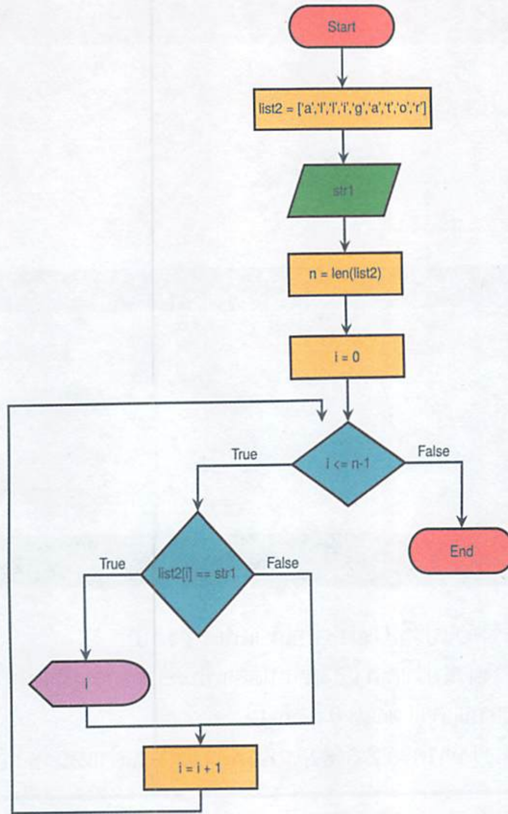
บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 เพื่อเก็บเลข 1 ถึง 10
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list1
บรรทัดที่ 3 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 4-5 : ถ้า i หารด้วย 2 ลงตัว ให้พิมพ์ค่า list1 ในตำแหน่งที่ i ออกมา

เฉลยข้อที่ 38. แนวคิดเดียวกับข้อ 37

เฉลยข้อที่ 39. แนวคิดเดียวกับข้อ 37

12

เฉลยข้อที่ 40.



Input :

```
1 list2 = ['a', 'l', 'l', 'i', 'g', 'a', 't', 'o', 'r']
2 str1 = input('Please insert str1: ')
3 n = len(list2)
4 for i in range(n):
5     if list2[i] == str1:
6         print(i)
```

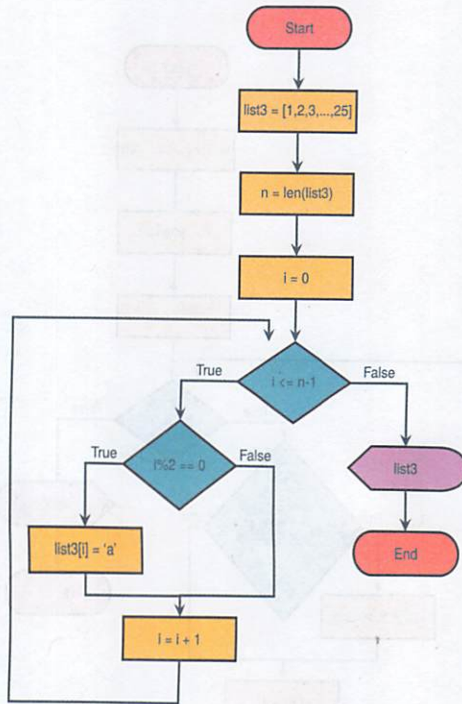
Output :

```
Please insert str1: a
0
5
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list2 เพื่อเก็บค่าอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 3 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list2
บรรทัดที่ 4 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 5-6 : ถ้าสมาชิกของ list2 ในตำแหน่งที่ i มีค่าเท่ากับ str1 ให้พิมพ์ค่า i ออกมา

เฉลยข้อที่ 41.



Input :

```
1 list3 = [x for x in range(1, 26)]
2 n = len(list3)
3 for i in range(n):
4     if i%2 == 0:
5         list3[i] = 'a'
6 print(list3)
```

Output :

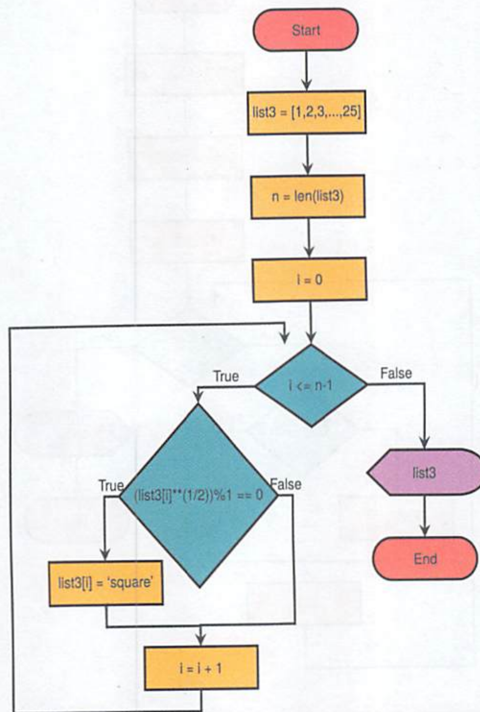
```
['a', 2, 'a', 4, 'a', 6, 'a', 8, 'a', 10, 'a', 12, 'a',
14, 'a', 16, 'a', 18, 'a', 20, 'a', 22, 'a', 24, 'a']
```


เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list3 เพื่อเก็บเลข 1 ถึง 25
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list3
บรรทัดที่ 4 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 5-6 : ถ้า i เป็นเลขคู่ ให้เปลี่ยนสมาชิกของ list3 ในตำแหน่งที่ i เป็น 'a'
บรรทัดที่ 7 : พิมพ์ค่า list3 ออกมา

เฉลยข้อที่ 42. แนวคิดเดียวกับข้อ 41

เฉลยข้อที่ 43.



Input :

```
1 list3 = [x for x in range(1, 26)]
2 n = len(list3)
3 for i in range(n):
4     if (list3[i]**(1/2))%1 == 0:
5         list3[i] = 'square'
6 print(list3)
```

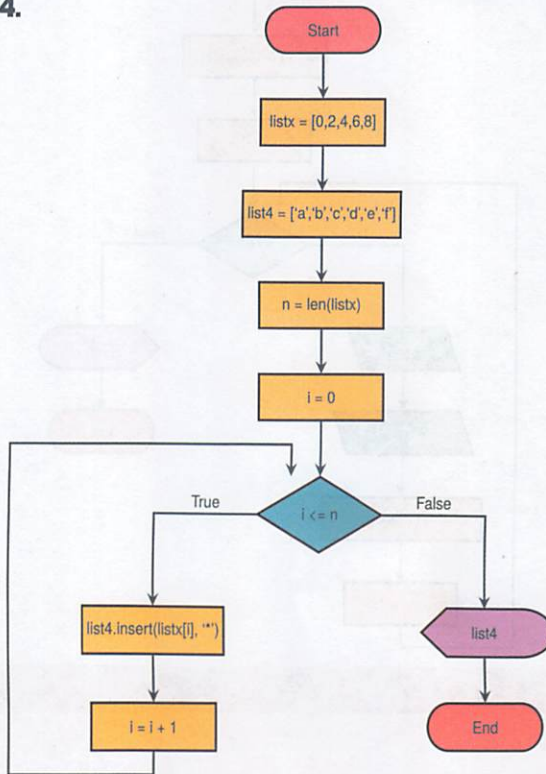
Output :

```
['square', 2, 3, 'square', 5, 6, 7, 8, 'square', 10,
11, 12, 13, 14, 15, 'square', 17, 18, 19, 20, 21, 22,
23, 24, 'square']
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list3 เพื่อเก็บเลข 1 ถึง 25
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list3
บรรทัดที่ 4 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 5-6 : ถ้ารากที่สองของสมาชิกในตำแหน่งที่ i ของ list3 เป็นจำนวนเต็ม ให้เปลี่ยนค่าเป็น 'square'
บรรทัดที่ 7 : พิมพ์ค่า list3 ออกมา

เฉลยข้อที่ 44.



Input :

```
1 listx = [0, 2 ,4 ,6, 8]
2 list4 = ['a', 'b', 'c', 'd', 'e', 'f']
3 n = len(listx)
4 for i in range(n):
5     list4.insert(listx[i], '*')
6 print(list4)
```

12

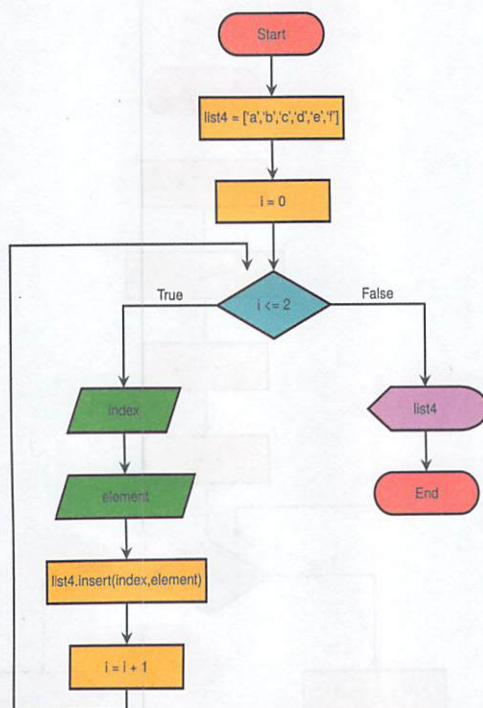
Output :

```
['*', 'a', '*', 'b', '*', 'c', '*', 'd', '*', 'e', 'f']
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ listx เพื่อเก็บค่าตัวเลขตามที่โจทย์กำหนด
บรรทัดที่ 2 : สร้างตัวแปรชื่อ list4 เพื่อเก็บค่าอักขระตามที่โจทย์กำหนด
บรรทัดที่ 3 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ listx
บรรทัดที่ 4 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 5 : นำ '*' ไป insert ใน list4 ณ ตำแหน่งที่ i ของ listx
บรรทัดที่ 6 : พิมพ์ค่า list4 ออกมา

เฉลยข้อที่ 45.



Input :

```
1 list4 = ['a', 'b', 'c', 'd', 'e', 'f']
2 for i in range(3):
3     index = int(input('Please insert index: '))
4     element = input('Please insert element: ')
5     list4.insert(index, element)
6 print(list4)
```


Output :

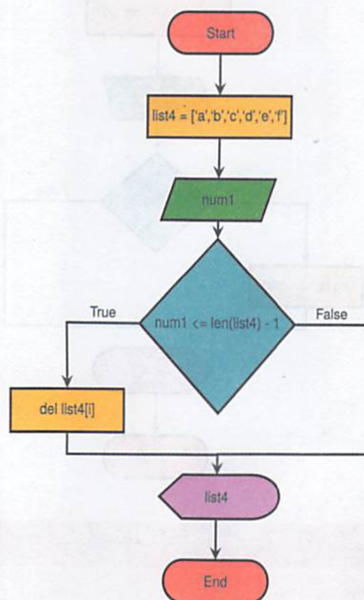
```
Please insert index: 2
Please insert element: -
Please insert index: 4
Please insert element: *
Please insert index: 5
Please insert element: ^
['a', 'b', '-', 'c', '*', '^', 'd', 'e', 'f']
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list4 เพื่อเก็บอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-5 ซ้ำ 3 ครั้ง
บรรทัดที่ 3 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปร index
บรรทัดที่ 4 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปร element
บรรทัดที่ 5 : นำ element ไป insert ใน list4 ณ ตำแหน่ง index
บรรทัดที่ 6 : พิมพ์ค่า list4 ออกมา

เฉลยข้อที่ 46. แนวคิดเดียวกับข้อ 45

เฉลยข้อที่ 47.



Input :

```
1 list4 = ['a', 'b', 'c', 'd', 'e', 'f']
2 num1 = int(input('Please insert num1: '))
3 if num1 <= len(list4) - 1:
4     del list4[num1]
5 print(list4)
```

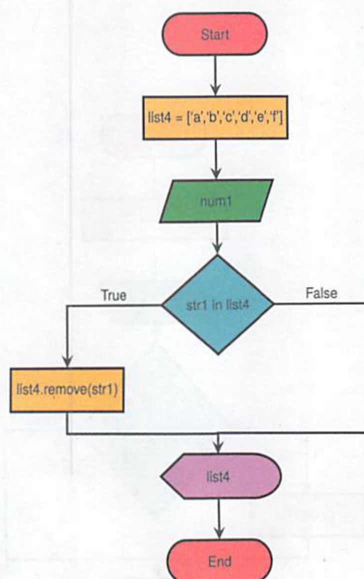
Output :

```
Please insert num1: 3
['a', 'b', 'c', 'e', 'f']
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list4 เพื่อเก็บค่าอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ num1
บรรทัดที่ 3-4 : ถ้า num1 มีค่าน้อยกว่าหรือเท่ากับ ความยาวของ list4 – 1 ให้ลบ list4 ในตำแหน่งที่ num1 ทิ้ง
บรรทัดที่ 5 : พิมพ์ค่า list4 ออกมา

เฉลยข้อที่ 48.



Input :

```
1 list4 = ['a', 'b', 'c', 'd', 'e', 'f']
2 str1 = input('Please insert str1: ')
3 if str1 in list4:
4     list4.remove(str1)
5 print(list4)
```

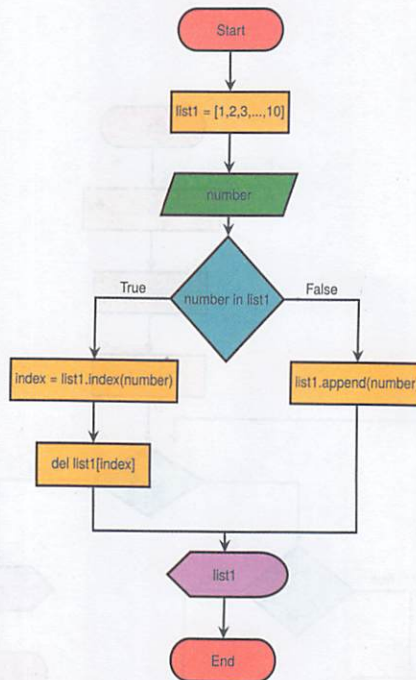
Output :

```
Please insert str1: d
['a', 'b', 'c', 'e', 'f']
```


เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list4 เพื่อเก็บค่าอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 3-4 : ถ้า str1 เป็นสมาชิกของ list4 ให้ลบ str1 ออกจาก list4
บรรทัดที่ 5 : พิมพ์ค่า list4 ออกมา

เฉลยข้อที่ 49.



Input :

```
1 list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
2 number = int(input('Please insert number: '))
3 if number in list1:
4     index = list1.index(number)
5     del list1[index]
6 else:
7     list1.append(number)
8 print(list1)
```


Output :

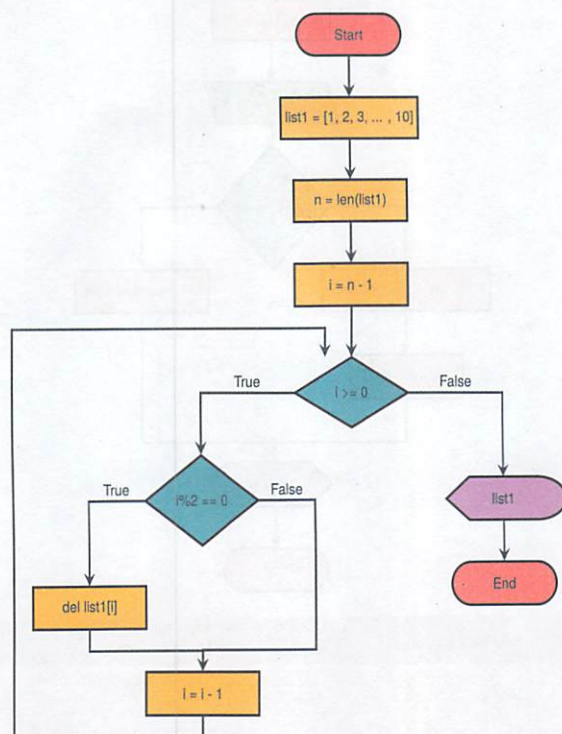
```
Please insert number: 3
[1, 2, 4, 5, 6, 7, 8, 9, 10]
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 เพื่อเก็บเลข 1 ถึง 10
บรรทัดที่ 2 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 3 : ถ้า number เป็นสมาชิกของ list1 ให้ทำบรรทัดที่ 4-5
บรรทัดที่ 4-5 : สร้างตัวแปรชื่อ index เพื่อเก็บค่าตำแหน่งของ number ใน list1 และลบ list1 ในตำแหน่งที่ index
บรรทัดที่ 6-7 : ถ้า number ไม่เป็นสมาชิกของ list1 ให้ append ใน list1
บรรทัดที่ 8 : พิมพ์ค่า list1 ออกมา

เฉลยข้อที่ 50. แนวคิดเดียวกับข้อ 49

เฉลยข้อที่ 51.



Input :

```
1 list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
2 n = len(list1)
3 for i in range(n-1, -1, -1):
4     if i%2 == 0:
5         del list1[i]
6 print(list1)
```

Output :

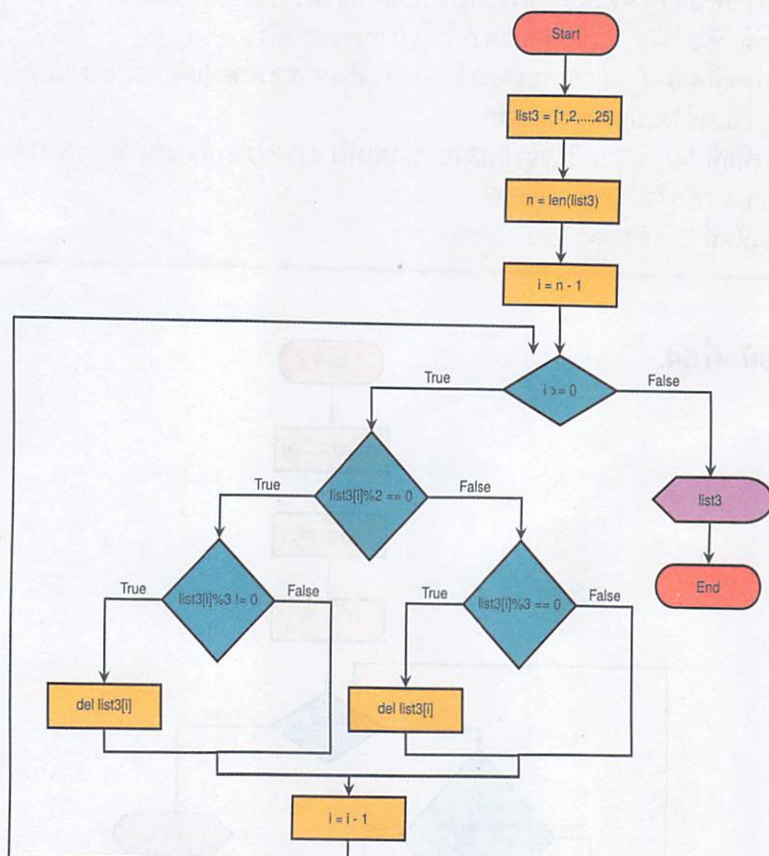
```
[2, 4, 6, 8, 10]
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 เพื่อเก็บเลข 1 ถึง 10
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list1
บรรทัดที่ 3 : พิจารณาค่า i ตั้งแต่ n-1 ถึง 0 แบบถอยหลัง
บรรทัดที่ 4-5 : ถ้า i เป็นเลขคู่ให้ลบสมาชิกของ list1 ในตำแหน่งที่ i ทิ้ง
บรรทัดที่ 6 : พิมพ์ค่า list1 ออกมา

เฉลยข้อที่ 52. แนวคิดเดียวกับข้อ 51

เฉลยข้อที่ 53.



Input :

```
list3 = [x for x in range(1, 26)]
n = len(list3)
for i in range(n-1, -1, -1):
    if list3[i]%2 == 0:
        if list3[i]%3 != 0:
            del list3[i]
    else:
        if list3[i]%3 == 0:
            del list3[i]
print(list3)
```

Output :

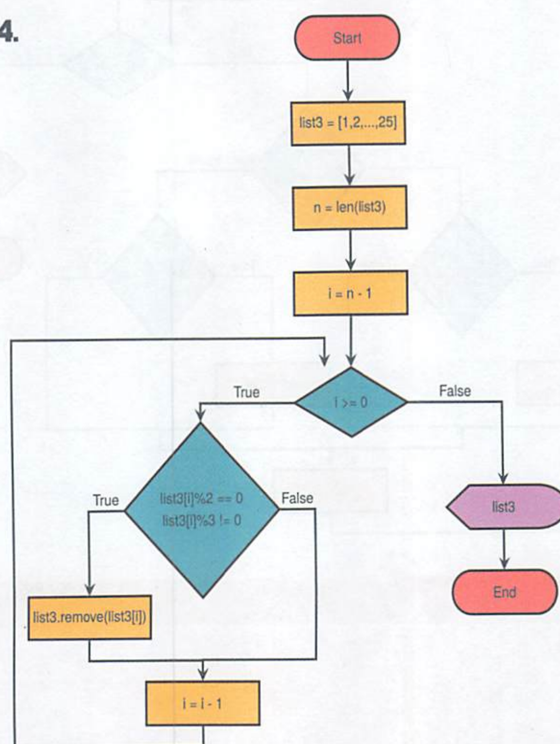
```
[2, 3, 5, 6, 7, 8, 10, 11, 12, 13, 14, 15, 17, 18, 19,
20, 21, 22, 23, 24]
```

12

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list3 เพื่อเก็บเลข 1 ถึง 25
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list3
บรรทัดที่ 3 : พิจารณาค่า i ตั้งแต่ n-1 ถึง 0 แบบถอยหลัง
บรรทัดที่ 4-6 : ถ้าสมาชิกของ list3 ในตำแหน่งที่ i หาร 2 ลงตัว แต่หารด้วย 3 ไม่ลงตัว ให้ลบสมาชิกตัวนั้นออกจาก list3
บรรทัดที่ 7-9 : ถ้าสมาชิกของ list3 ในตำแหน่งที่ i หาร 2 ไม่ลงตัว แต่หารด้วย 3 ลงตัว ให้ลบสมาชิกตัวนั้นออกจาก list3
บรรทัดที่ 10 : พิมพ์ค่า list3 ออกมา

เฉลยข้อที่ 54.



Input :

```
list3 = [x for x in range(1, 26)]
n = len(list3)
for i in range(n-1, -1, -1):
    if (list3[i]%2 == 0) and (list3[i]%3 != 0):
        list3.remove(list3[i])
print(list3)
```

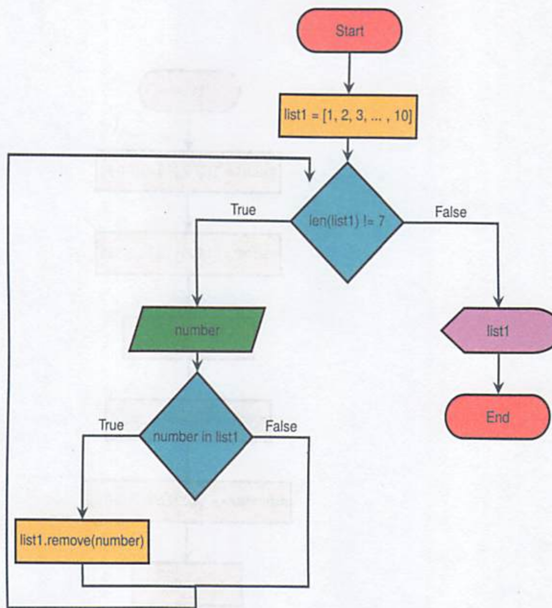
Output :

```
[1, 3, 5, 6, 7, 9, 11, 12, 13, 15, 17, 18, 19, 21, 23,
24, 25]
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list3 เพื่อเก็บเลข 1 ถึง 25
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list3
บรรทัดที่ 3 : พิจารณาค่า i ตั้งแต่ n-1 ถึง 0 แบบถอยหลัง
บรรทัดที่ 4-5 : ถ้าสมาชิกของ list3 ในตำแหน่งที่ i หารด้วย 2 ลงตัว แต่หารด้วย 3 ไม่ลงตัว ให้ลบสมาชิกตัวนั้นออกจาก list3
บรรทัดที่ 6 : พิมพ์ค่า list3 ออกมา

เฉลยข้อที่ 55.



Input :

```
list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
while len(list1) != 7:
    number = int(input('Please insert number: '))
    if number in list1:
        list1.remove(number)
print(list1)
```

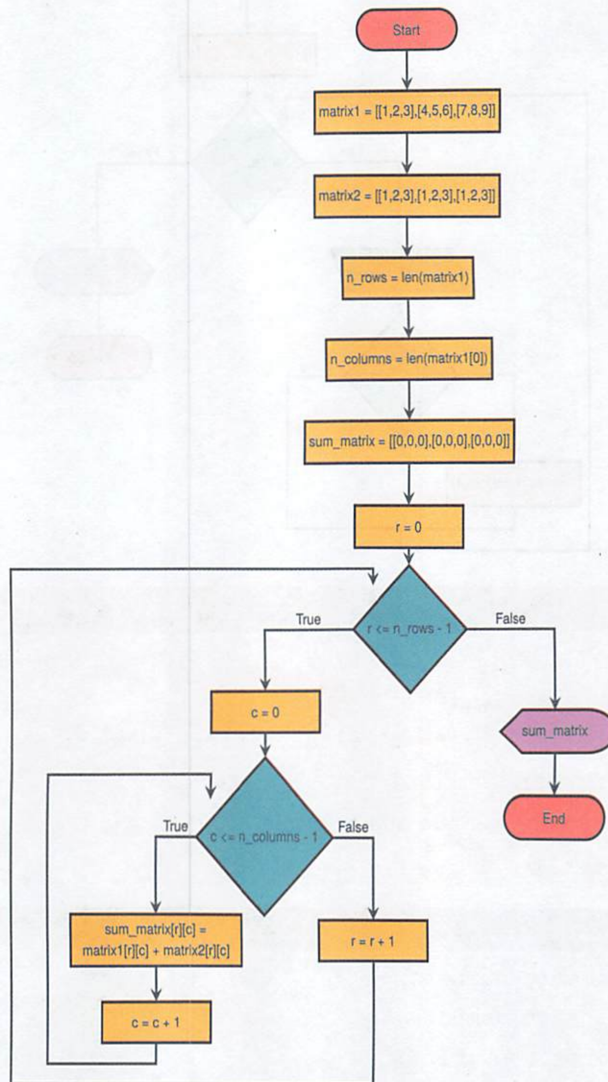
Output :

```
Please insert number: 3
Please insert number: 3
Please insert number: 5
Please insert number: 7
[1, 2, 4, 6, 8, 9, 10]
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list3 เพื่อเก็บเลข 1 ถึง 10
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-5 ซ้ำ จนกว่าความยาวของ list1 จะน้อยกว่าหรือเท่ากับ 7
บรรทัดที่ 3 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 4-5 : ถ้า number ที่รับมาเป็นสมาชิกของ list1 ให้ลบ number ออกจาก list1
บรรทัดที่ 6 : พิมพ์ค่า list1 ออกมา

เฉลยข้อที่ 56.



12

Input :

```
1 matrix1 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
2 matrix2 = [[1, 2, 3], [1, 2, 3], [1, 2, 3]]
3 n_rows = len(matrix1)
4 n_columns = len(matrix1[0])
5 sum_matrix = [[0, 0, 0], [0, 0, 0], [0, 0, 0]]
6 for r in range(n_rows):
7     for c in range(n_columns):
8         sum_matrix[r][c] = matrix1[r][c] + matrix2[r][c]
9 print(sum_matrix)
```

Output :

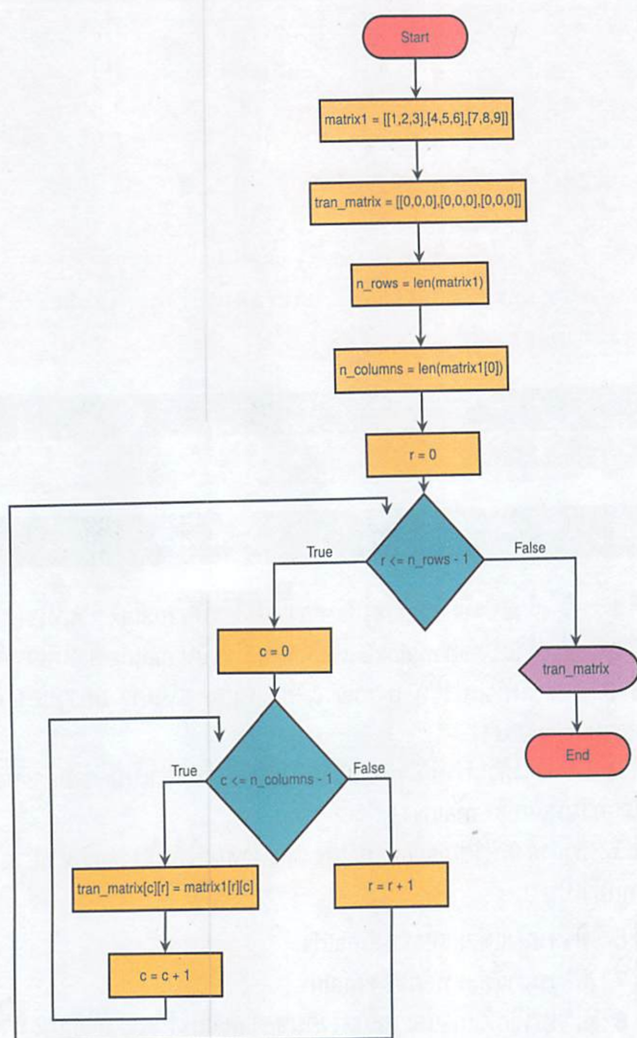
```
[[2, 4, 6], [5, 7, 9], [8, 10, 12]]
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ matrix1 และกำหนดให้เป็น matrix ตามที่โจทย์กำหนด
บรรทัดที่ 2 : สร้างตัวแปรชื่อ matrix2 และกำหนดให้เป็น matrix ตามที่โจทย์กำหนด
บรรทัดที่ 3 : สร้างตัวแปรชื่อ n_row และกำหนดค่าเป็นความยาวของ matrix1 (จำนวนแถวของ matrix1)
บรรทัดที่ 4 : สร้างตัวแปรชื่อ n_column และกำหนดค่าเป็นความยาวของ matrix1[0] (จำนวนหลักของ matrix1)
บรรทัดที่ 5 : สร้างตัวแปรชื่อ sum_matrix และกำหนดให้เป็น matrix ขนาด 3×3 ที่มีสมาชิกทุกตัวเป็น 0
บรรทัดที่ 6 : พิจารณาแต่ละแถวของ matrix
บรรทัดที่ 7 : พิจารณาแต่ละหลักของ matrix
บรรทัดที่ 8 : นำสมาชิกที่แถวและหลักตรงกันของ matrix1 และ matrix2 มาบวกกันแล้วไปรวมไว้ใน sum_matrix
บรรทัดที่ 9 : พิมพ์ค่า sum_matrix ออกมา

เฉลยข้อที่ 57. แนวคิดเดียวกับข้อ 56

เฉลยข้อที่ 58.



12

Input :

```
matrix1 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
tran_matrix = [[0, 0, 0], [0, 0, 0], [0, 0, 0]]
n_rows = len(matrix1)
n_columns = len(matrix1[0])
for r in range(n_rows):
    for c in range(n_columns):
        tran_matrix[c][r] = matrix1[r][c]
print(tran_matrix)
```

Output :

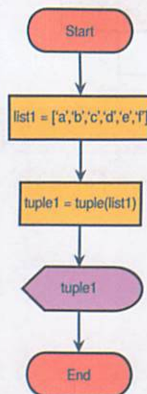
```
[[1, 4, 7], [2, 5, 8], [3, 6, 9]]
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ matrix1 และกำหนดให้เป็น matrix ตามที่โจทย์กำหนด
บรรทัดที่ 2 : สร้างตัวแปรชื่อ trans_matrix และกำหนดให้เป็น matrix ขนาด 3×3 ที่มีสมาชิกทุกตัวเป็น 0
บรรทัดที่ 3 : สร้างตัวแปรชื่อ n_row และกำหนดค่าเป็นความยาวของ matrix1 (จำนวนแถวของ matrix1)
บรรทัดที่ 4 : สร้างตัวแปรชื่อ n_column และกำหนดค่าเป็นความยาวของ matrix1[0] (จำนวนหลักของ matrix1)
บรรทัดที่ 5 : พิจารณาแต่ละแถวของ matrix ในชื่อตัวแปร r
บรรทัดที่ 6 : พิจารณาแต่ละหลักของ matrix ในชื่อตัวแปร c
บรรทัดที่ 7 : นำสมาชิกในแถว r และหลัก c ของ matrix1 ไปใส่ในตำแหน่งแถว c และหลัก r ของ tran_matrix
บรรทัดที่ 8 : พิมพ์ tran_matrix ออกมา

เฉลยข้อที่ 59. แนวคิดเดียวกับข้อ 58

เฉลยข้อที่ 60.



12

Input :

```
1 list1 = ['a', 'b', 'c', 'd', 'e', 'f']
2 tuple1 = tuple(list1)
3 print(tuple1)
```

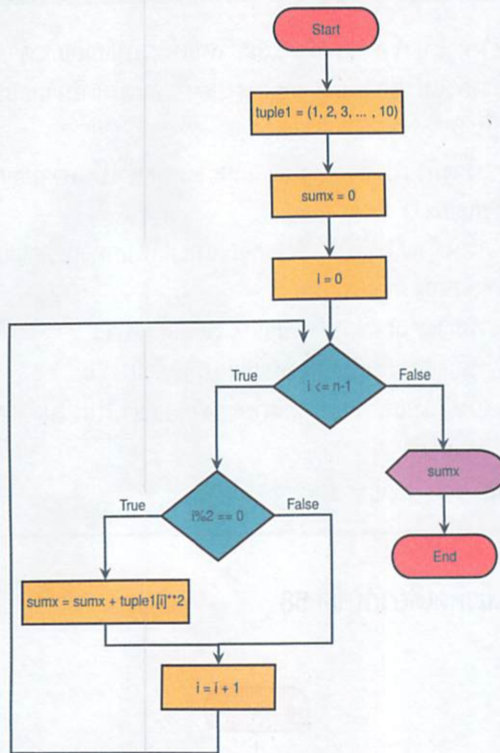
Output :

```
('a', 'b', 'c', 'd', 'e', 'f')
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 เพื่อเก็บอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : สร้างตัวแปรชื่อ tuple1 เพื่อแปลง list1 ให้กลายเป็น tuple
บรรทัดที่ 3 : พิมพ์ค่า tuple1 ออกมา

เฉลยข้อที่ 61.



Input :

```
1 tuple1 = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
2 sumx = 0
3 n = len(tuple1)
4 for i in range(n):
5     if i%2 == 0:
6         sumx = sumx + tuple1[i]**2
7 print(sumx)
```

Output :

```
165
```

เฉลยละเอียด

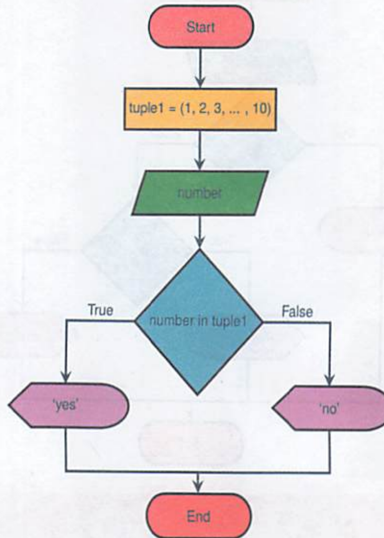
บรรทัดที่ 1 : สร้างตัวแปรชื่อ tuple1 เพื่อเก็บค่าเลข 1 ถึง 10
บรรทัดที่ 2 : สร้างตัวแปรชื่อ sumx และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ tuple1
บรรทัดที่ 4 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 5-6 : ถ้า i เป็นเลขคู่ นำสมาชิกของ tuple1 ตำแหน่งที่ i มายกกำลังสอง แล้วไปบวกเพิ่มที่ sumx
บรรทัดที่ 7 : พิมพ์ค่า sumx ออกมา

เฉลยข้อที่ 62. แนวคิดเดียวกับข้อ 61

เฉลยข้อที่ 63. แนวคิดเดียวกับข้อ 17

เฉลยข้อที่ 64. แนวคิดเดียวกับข้อ 10 และ 17

เฉลยข้อที่ 65.



Input :

```
1 tuple1 = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
2 number = int(input('Please insert number: '))
3 if number in tuple1:
4     print('yes')
5 else:
6     print('no')
```

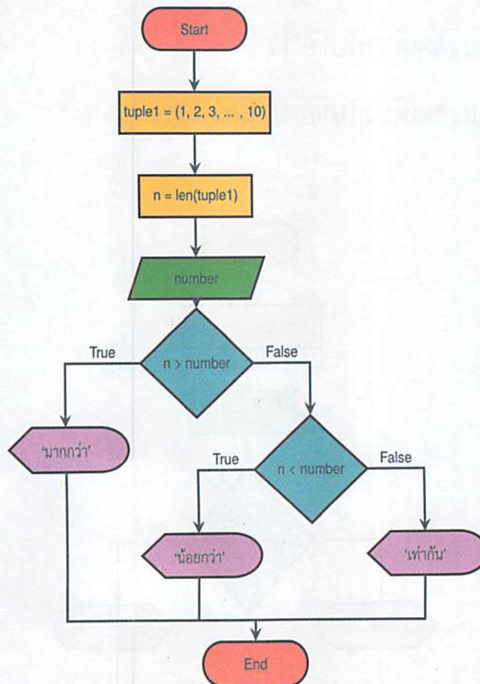
Output :

```
Please insert number: 7 yes
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ tuple1 เพื่อเก็บเลข 1 ถึง 10
บรรทัดที่ 2 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 3-4 : ถ้า number เป็นสมาชิกของ tuple1 ให้พิมพ์ 'yes'
บรรทัดที่ 5-6 : ถ้า number ไม่เป็นสมาชิกของ tuple1 ให้พิมพ์ 'no'

เฉลยข้อที่ 66.



Input :

```
1 tuple1 = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
2 n = len(tuple1)
3 number = int(input('Please insert number: '))
4 if n > number:
5     print('มากกว่า')
6 elif n < number:
7     print('น้อยกว่า')
8 else:
9     print('เท่ากับ')
```

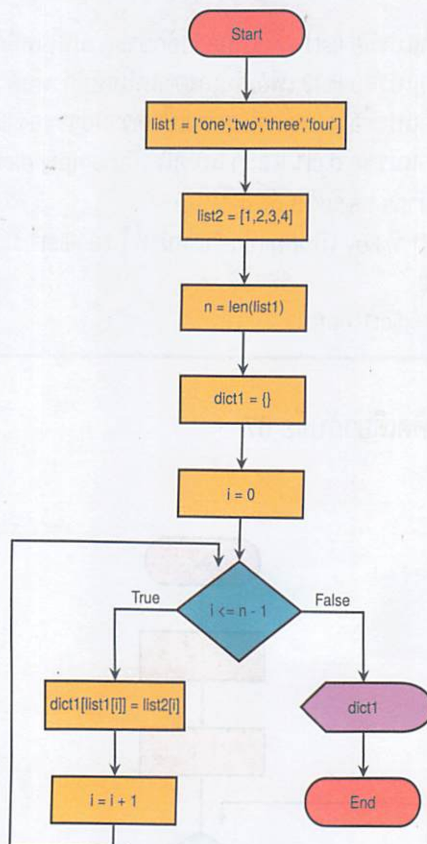
Output :

```
Please insert number: 7
มากกว่า
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ tuple1 เพื่อเก็บเลข 1 ถึง 10
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ tuple1
บรรทัดที่ 3 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 4-5 : ถ้า n มากกว่า number ให้พิมพ์ 'มากกว่า'
บรรทัดที่ 6-7 : ถ้า n น้อยกว่า number ให้พิมพ์ 'น้อยกว่า'
บรรทัดที่ 8-9 : ถ้า n เท่ากับ number ให้พิมพ์ 'เท่ากับ'

เฉลยข้อที่ 67.



Input :

```
list1 = ['one', 'two', 'three', 'four']
list2 = [1, 2, 3, 4]
n = len(list1)
dict1 = {}
for i in range(n):
    dict1[list1[i]] = list2[i]
print(dict1)
```

Output :

```
{'one': 1, 'two': 2, 'three': 3, 'four': 4}
```

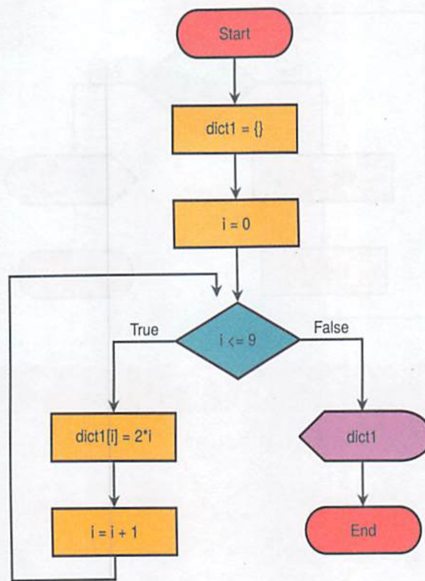
12

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 เพื่อเก็บสายอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : สร้างตัวแปรชื่อ list2 เพื่อเก็บเลขตามที่โจทย์กำหนด
บรรทัดที่ 3 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list1
บรรทัดที่ 4 : สร้างตัวแปรชื่อ dict1 และกำหนดให้เป็น empty dict
บรรทัดที่ 5 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 6 : ให้ dict1 ที่ key เป็นสมาชิกตำแหน่งที่ i ของ list1 มี value เป็นสมาชิกตำแหน่งที่ i ของ list2
บรรทัดที่ 7 : พิมพ์ค่า dict1 ออกมา

เฉลยข้อที่ 68. แนวคิดเดียวกับข้อ 67

เฉลยข้อที่ 69.



Input :

```python
dict1 = {}
for i in range(10):
    dict1[i] = 2*i
print(dict1)
```

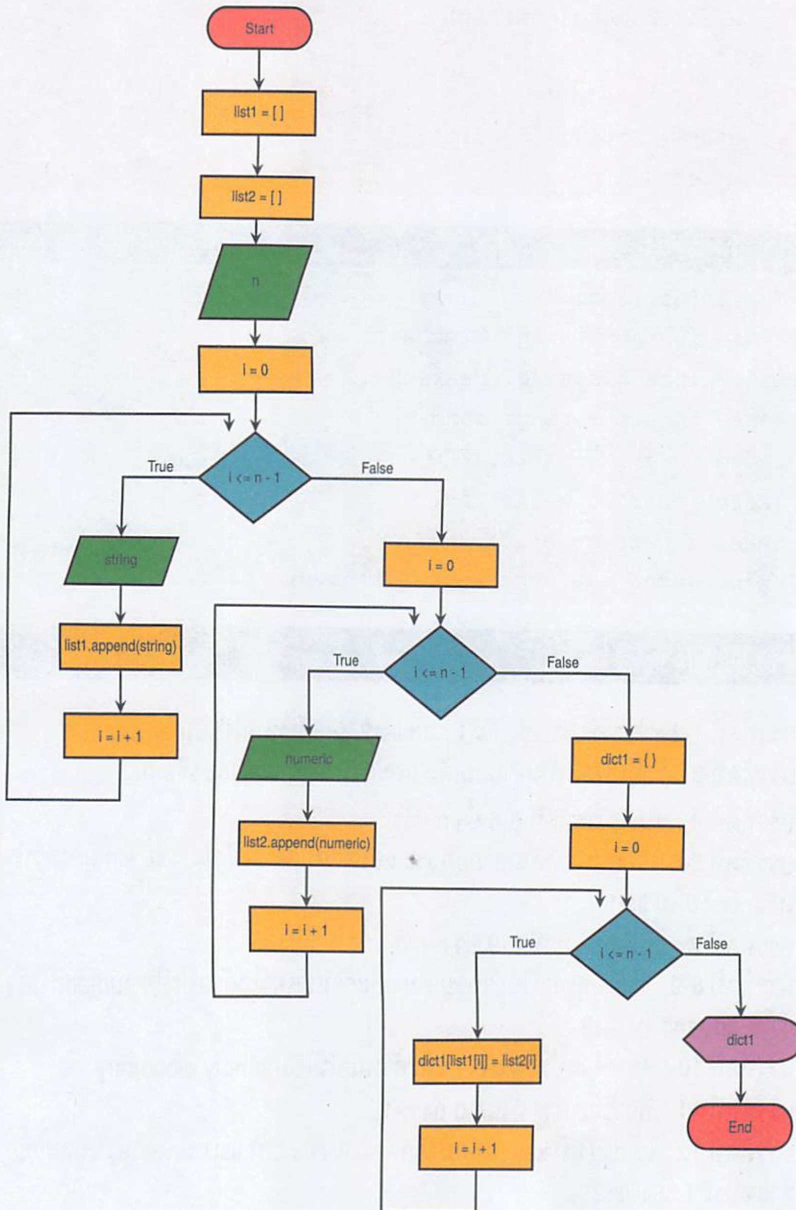
Output :

```
{0: 0, 1: 2, 2: 4, 3: 6, 4: 8, 5: 10, 6: 12, 7: 14, 8:
16, 9: 18}
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ dict1 และกำหนดให้เป็น empty dictionary
บรรทัดที่ 2 : พิจารณาค่า i ตั้งแต่ 0 ถึง 9
บรรทัดที่ 3 : ให้ dict1 ที่ key เป็น i มี value เป็น 2*i
บรรทัดที่ 4 : พิมพ์ค่า dict1 ออกมา

เฉลยข้อที่ 70.



12

Input :

```
1  list1 = []
2  list2 = []
3  n = int(input('Please insert n: '))
4  for i in range(n):
5      string = input('Please insert string: ')
6      list1.append(string)
7  for i in range(n):
8      numeric = int(input('Please insert numeric: '))
9      list2.append(numeric)
10 dict1 = {}
11 for i in range(n):
12     dict1[list1[i]] = list2[i]
13 print(dict1)
```

Output :

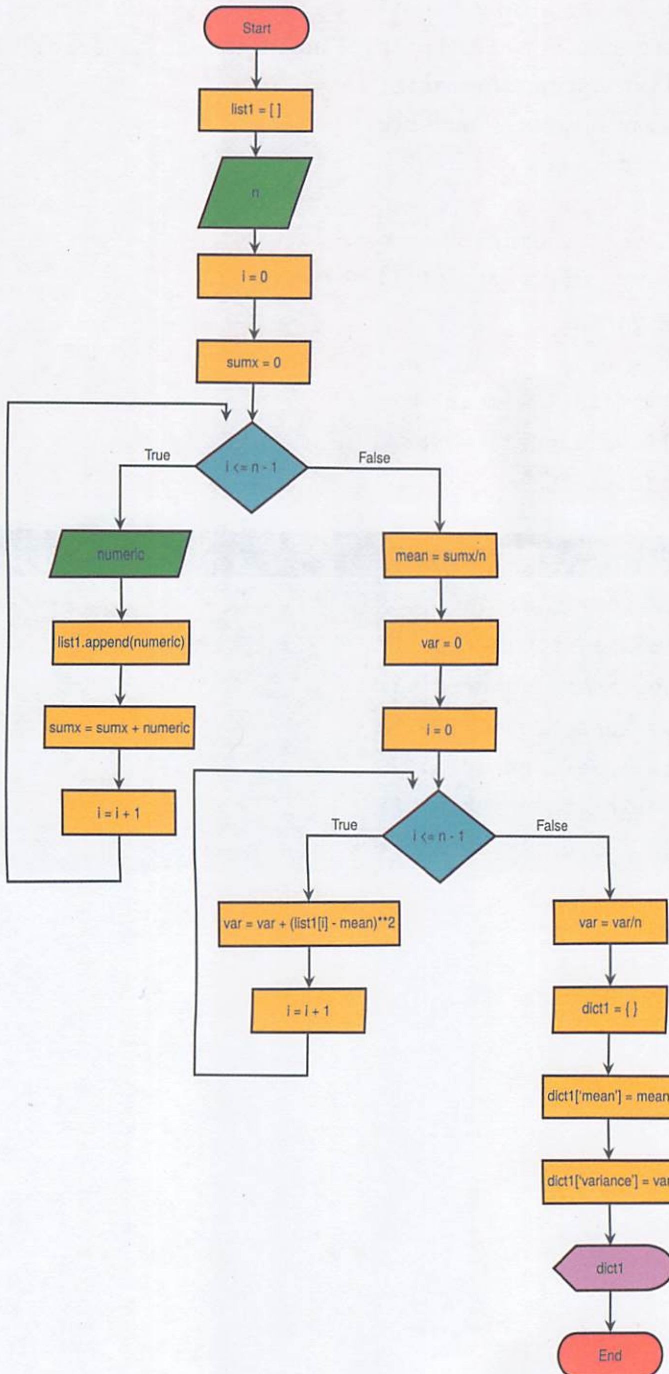
```
Please insert n: 3
Please insert string: armchair
Please insert string: basketball
Please insert string: candy
Please insert numeric: 700
Please insert numeric: 500
Please insert numeric: 3
{'armchair': 700, 'basketball': 500, 'candy': 3}
```


เฉลยละเอียด

บรรทัดที่ 1-2 : สร้างตัวแปรชื่อ list1 และ list2 และกำหนดให้เป็น empty list
บรรทัดที่ 3 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ n
บรรทัดที่ 4 : วนทำบรรทัดที่ 5-6 ซ้ำ n ครั้ง
บรรทัดที่ 5-6 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ string แล้วนำไป append ใน list1
บรรทัดที่ 7 : วนทำบรรทัดที่ 8-9 ซ้ำ n ครั้ง
บรรทัดที่ 8-9 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ numeric แล้วนำไป append ใน list2
บรรทัดที่ 10 : สร้างตัวแปรชื่อ dict1 และกำหนดให้เป็น empty dictionary
บรรทัดที่ 11 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 12 : ให้ dict1 ที่ key เป็นสมาชิกตำแหน่งที่ i ของ list1 มี value เป็นสมาชิกตำแหน่งที่ i ของ list2
บรรทัดที่ 13 : พิมพ์ค่า dict1 ออกมา

เฉลยข้อที่ 71. แนวคิดเดียวกับข้อ 70

เฉลยข้อที่ 72.



12

Input :

```
list1 = []
n = int(input('Please insert n: '))
sumx = 0
for i in range(n):
    numeric = float(input('Please insert numeric: '))
    list1.append(numeric)
    sumx = sumx + numeric
mean = sumx/n
var = 0
for i in range(n):
    var = var + (list1[i] - mean)**2
var = var/n
dict1 = {}
dict1['mean'] = mean
dict1['variance'] = var
print(dict1)
```

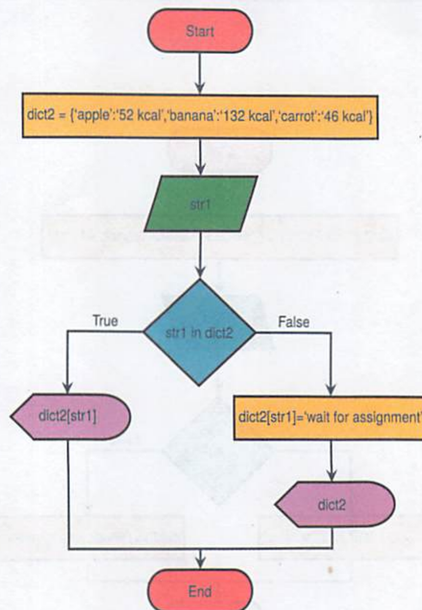
Output :

```
Please insert n: 5
Please insert numeric: 1
Please insert numeric: 3
Please insert numeric: 5
Please insert numeric: 7
Please insert numeric: 9
{'mean': 5.0, 'variance': 8.0}
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 และกำหนดให้เป็น empty list
บรรทัดที่ 2 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ n
บรรทัดที่ 3 : สร้างตัวแปรชื่อ sumx และกำหนดค่าเท่ากับ 0
บรรทัดที่ 4 : วนทำบรรทัดที่ 5-7 ซ้ำ n ครั้ง
บรรทัดที่ 5-7 : รับอินพุตที่เป็นจำนวนจริง และกำหนดค่าใส่ตัวแปรชื่อ numeric แล้วนำไป append ใน list1 และนำไปบวกเพิ่มที่ sumx
บรรทัดที่ 8 : สร้างตัวแปรชื่อ mean เพื่อคำนวณค่าเฉลี่ยของข้อมูล
บรรทัดที่ 9 : สร้างตัวแปรชื่อ var และกำหนดค่าเท่ากับ 0
บรรทัดที่ 10 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 11 : นำกำลังสองของผลต่างระหว่างสมาชิกตำแหน่งที่ i ของ list1 กับค่า mean ไปบวกเพิ่มที่ var
บรรทัดที่ 12 : หารเฉลี่ยผลบวกที่ได้ แล้วเก็บไว้ในตัวแปร var
บรรทัดที่ 13 : สร้างตัวแปรชื่อ dict1 และกำหนดให้เป็น empty dictionary
บรรทัดที่ 14 : กำหนดให้ dict1 มี key เป็น 'mean' และมี value เป็น mean
บรรทัดที่ 15 : กำหนดให้ dict1 มี key เป็น 'var' และมี value เป็น var
บรรทัดที่ 16 : พิมพ์ค่า dict1 ออกมา

เฉลยข้อที่ 73.



12

Input :

```
1 dict2 = {'apple': '52 kcal', 'banana': '132 kcal', '
    ↪ carrot': '46 kcal'}
2 str1 = input('Please insert str1: ')
3 if str1 in dict2:
4     print(dict2[str1])
5 else:
6     dict2[str1] = 'wait for assignment'
7     print(dict2)
```

Output :

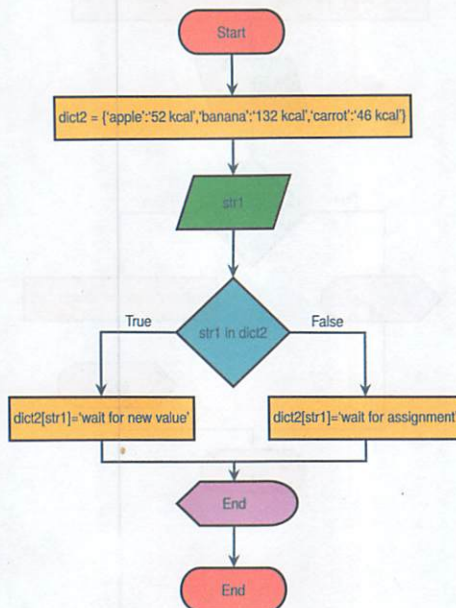
```
Please insert str1: apple
52 kcal
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ dict2 เพื่อเก็บสายอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 3-4 : ถ้า str1 เป็น key ใน dict2 ให้พิมพ์ value ของ str1 ใน dict2 ออกมา
บรรทัดที่ 5-7 : ถ้า str1 ไม่เป็น key ใน dict2 ให้สร้าง key ใหม่ชื่อ str1 และกำหนด value เป็น 'wait for assignment' และพิมพ์ค่า dict2 ออกมา

เฉลยข้อที่ 74.

12



Input :

```
1 dict2 = {'apple': '52 kcal', 'banana': '132 kcal', 'carrot': '46 kcal'}
2 str1 = input('Please insert str1: ')
3 if str1 in dict2:
4     dict2[str1] = 'wait for new value'
5 else:
6     dict2[str1] = 'wait for assignment'
7 print(dict2)
```

Output :

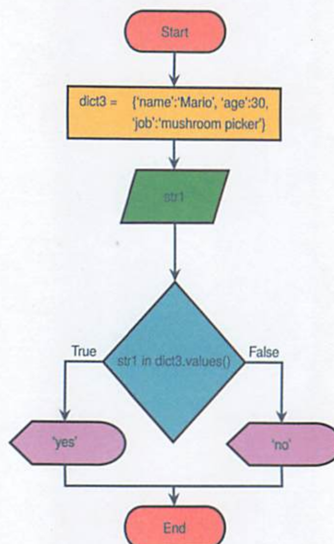
```
Please insert str1: xxx
{'apple': '52 kcal', 'banana': '132 kcal', 'carrot':
'46 kcal', 'xxx': 'wait for assignment'}
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ dict2 เพื่อเก็บสายอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 3-4 : ถ้า str1 เป็น key ใน dict2 ให้สร้าง key ใหม่ชื่อ str1 และกำหนด value เป็น 'wait for new value'
บรรทัดที่ 5-6 : ถ้า str1 ไม่เป็น key ใน dict2 ให้สร้าง key ใหม่ชื่อ str1 และกำหนด value เป็น 'wait for assignment'
บรรทัดที่ 7 : พิมพ์ค่า dict2 ออกมา

12

เฉลยข้อที่ 75.



Input :

```
1 dict3 = {'name': 'Mario', 'age': '30', 'job': 'mushroom picker'}
2 str1 = input('Please insert str1: ')
3 if str1 in dict3.values():
4     print('yes')
5 else:
6     print('no')
```

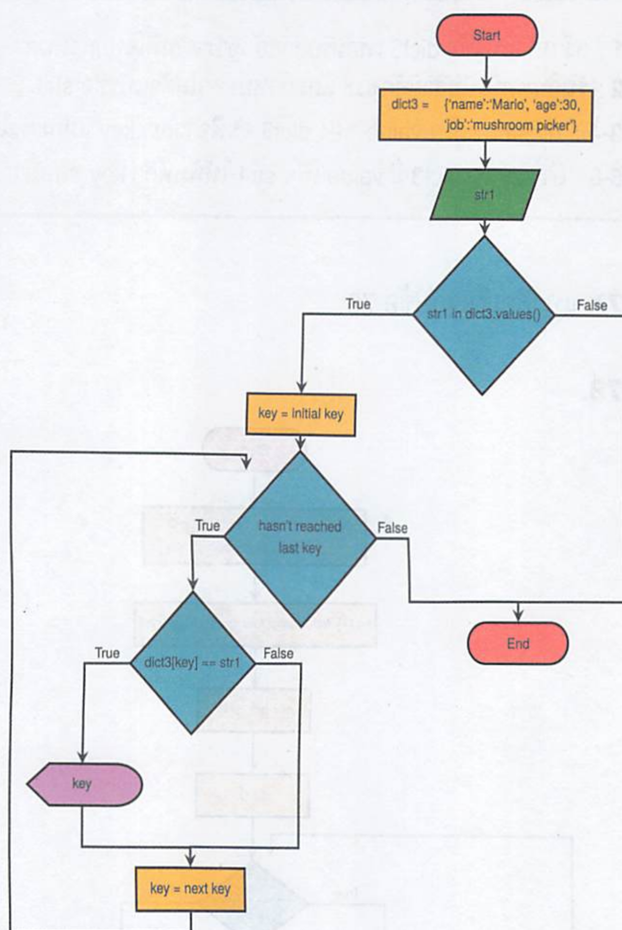
Output :

```
Please insert str1: 30
yes
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ dict3 เพื่อเก็บสายอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 3-4 : ถ้า str1 อยู่ใน value ของ dict3 ให้พิมพ์ 'yes'
บรรทัดที่ 5-6 : ถ้า str1 ไม่อยู่ใน value ของ dict3 ให้พิมพ์ 'no'

เฉลยข้อที่ 76.



12

Input :

```
dict3 = {'name': 'Mario', 'age': '30', 'job': 'mushroom
    ↪ picker'}
str1 = input('Please insert str1: ')
if str1 in dict3.values():
    for key in dict3:
        if dict3[key] == str1:
            print(key)
```

Output :

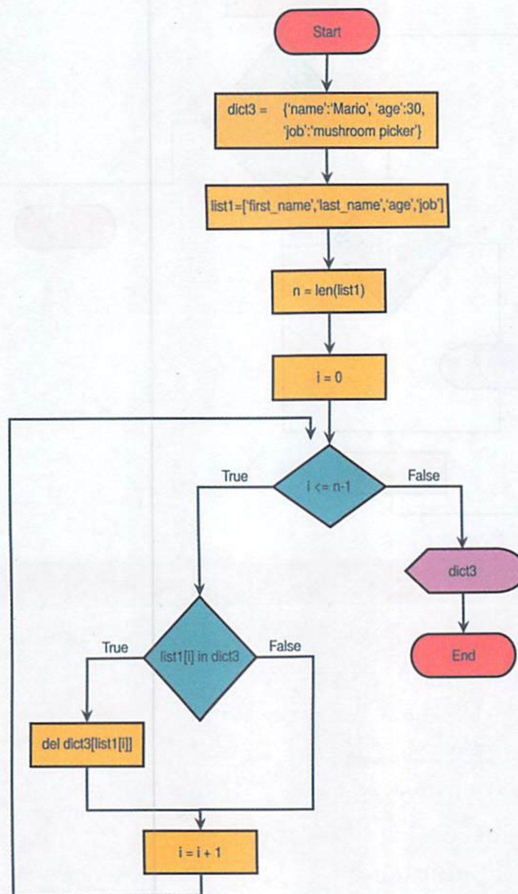
```
Please insert str1: mushroom picker
job
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ dict3 เพื่อเก็บสายอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 3-4 : ถ้า str1 อยู่ใน value ของ dict3 จะพิจารณา key แต่ละตัวใน dict3
บรรทัดที่ 5-6 : ถ้า key ใน dict3 มี value เป็น str1 ให้พิมพ์ค่า key ออกมา

เฉลยข้อที่ 77. แนวคิดเดียวกับข้อ 76

เฉลยข้อที่ 78.



Input :

```
1 dict3 = {'name': 'Mario', 'age': '30', 'job': 'mushroom
     ↪ picker'}
2 list1 = ['first_name', 'last_name', 'age', 'job']
3 n = len(list1)
4 for i in range(n):
5     if list1[i] in dict3:
6         del dict3[list1[i]]
7 print(dict3)
```

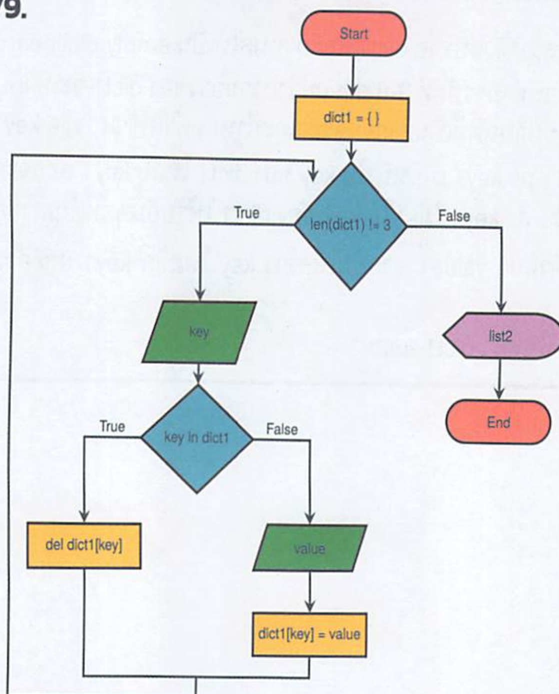
Output :

```
{'name': 'Mario'}
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ dict3 เพื่อเก็บสายอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : สร้างตัวแปรชื่อ list1 เพื่อเก็บสายอักระตามที่โจทย์กำหนด
บรรทัดที่ 3 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list1
บรรทัดที่ 4 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 5-6 : ถ้าสมาชิกตำแหน่งที่ i ของ list1 เป็นหนึ่งใน key ของ dict3 ให้ลบ key นั้นออกจาก dict3
บรรทัดที่ 7 : พิมพ์ค่า dict3 ออกมา

เฉลยข้อที่ 79.



Input :

```
1 dict1 = {}
2 while len(dict1) != 3:
3     key1 = input('Please insert key: ')
4     if key1 in dict1:
5         del dict1[key1]
6     else:
7         value1 = input('Please insert value: ')
8         dict1[key1] = value1
9 print(dict1)
```

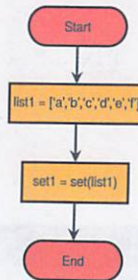
Output :

```
Please insert key: a
Please insert value: 1
Please insert key: a
Please insert key: b
Please insert value: 2
Please insert key: c
Please insert value: 3
Please insert key: d
Please insert value: 4
{'b': '2', 'c': '3', 'd': '4'}
```


เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ dict1 และกำหนดให้เป็น empty dictionary
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-8 ซ้ำจนกว่าความยาวของ dict1 จะเท่ากับ 3
บรรทัดที่ 3 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ key1
บรรทัดที่ 4-5 : ถ้า key1 เป็นหนึ่งใน key ของ dict1 ให้ลบ key1 ออกจาก dict1
บรรทัดที่ 6-8 : ถ้า key1 ไม่อยู่ใน key ของ dict1 ให้รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปร value1 จากนั้นให้สร้าง key ใหม่ชื่อ key1 และกำหนด value เป็น value1
บรรทัดที่ 9 : พิมพ์ค่า dict1 ออกมา

เฉลยข้อที่ 80.



Input :

```
1 list1 = ['a', 'b', 'c', 'd', 'e', 'f']
2 set1 = set(list1)
3 print(set1)
```

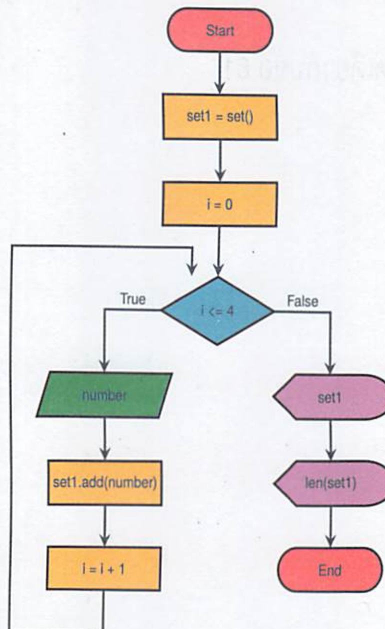
Output :

```
{'d', 'e', 'b', 'c', 'f', 'a'}
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 เพื่อเก็บอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : สร้างตัวแปรชื่อ set1 เพื่อแปลง list1 ให้กลายเป็น set
บรรทัดที่ 3 : พิมพ์ค่า set1 ออกมา

เฉลยข้อที่ 81.



Input :

```
1 set1 = set()
2 for i in range(5):
3     number = int(input('Please insert number: '))
4     set1.add(number)
5 print(set1)
6 print(len(set1))
```

Output :

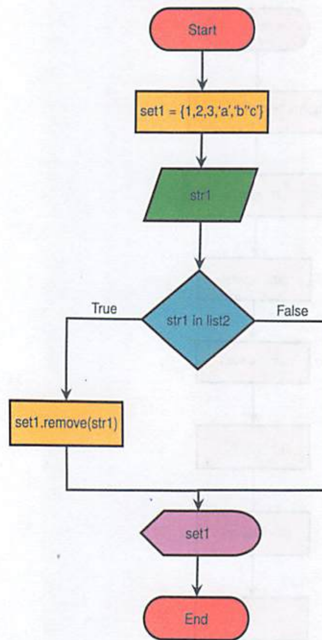
```
Please insert number: 1
Please insert number: 1
Please insert number: 2
Please insert number: 3
Please insert number: 5
{1, 2, 3, 5}
4
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ set1 และกำหนดให้เป็น empty set
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-4 ซ้ำ 5 ครั้ง
บรรทัดที่ 3 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 4 : นำ number ไปเพิ่มใน set1
บรรทัดที่ 5 : พิมพ์ค่า set1 ออกมา
บรรทัดที่ 6 : พิมพ์ความยาวของ set1 ออกมา

เฉลยข้อที่ 82. แนวคิดเดียวกับข้อ 81

เฉลยข้อที่ 83.



Input :

```
1 set1 = {1, 2, 3, 'a', 'b', 'c'}
2 str1 = input('Please insert str1: ')
3 if str1 in set1:
4     set1.remove(str1)
5 print(set1)
```

Output :

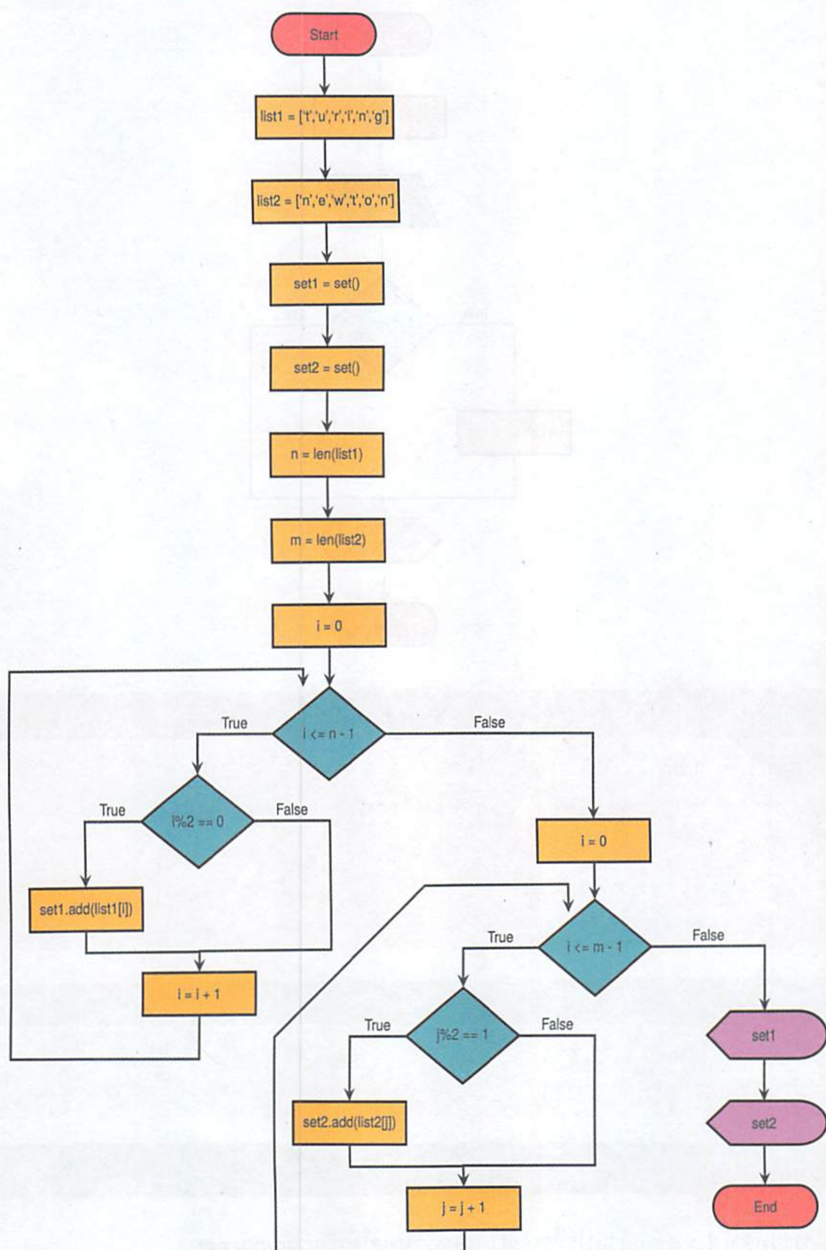
```
Please insert str1: a
{1, 2, 3, 'b', 'c'}
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ set1 และกำหนดให้เป็น empty set
บรรทัดที่ 2 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ str1
บรรทัดที่ 3-4 : ถ้า str1 เป็นสมาชิกของ set1 ให้ลบ str1 ออกจาก set1
บรรทัดที่ 5 : พิมพ์ค่า set1 ออกมา

12

เฉลยข้อที่ 84.


Start
list1 = ['t','u','r','i','n','g']
list2 = ['n','e','w','t','o','n']
set1 = set()
set2 = set()
n = len(list1)
m = len(list2)
i = 0
i <= n - 1
True
False
i%2 == 0
True
False
set1.add(list1[i])
i = i + 1
i = 0
i <= m - 1
True
False
j%2 == 1
True
False
set2.add(list2[j])
j = j + 1
set1
set2
End

Input :

```
1  list1 = ['t', 'u', 'r', 'i', 'n', 'g']
2  list2 = ['n', 'e', 'w', 't', 'o', 'n']
3  set1 = set()
4  set2 = set()
5  n = len(list1)
6  m = len(list2)
7  for i in range(n):
8      if i%2 == 0:
9          set1.add(list1[i])
10 for j in range(m):
11     if j%2 == 1:
12         set2.add(list2[j])
13 print('set1:', set1)
14 print('set2:', set2)
```

Output :

```
set1: {‘t’, ‘n’, ‘r’}
set2: {‘e’, ‘n’, ‘t’}
```

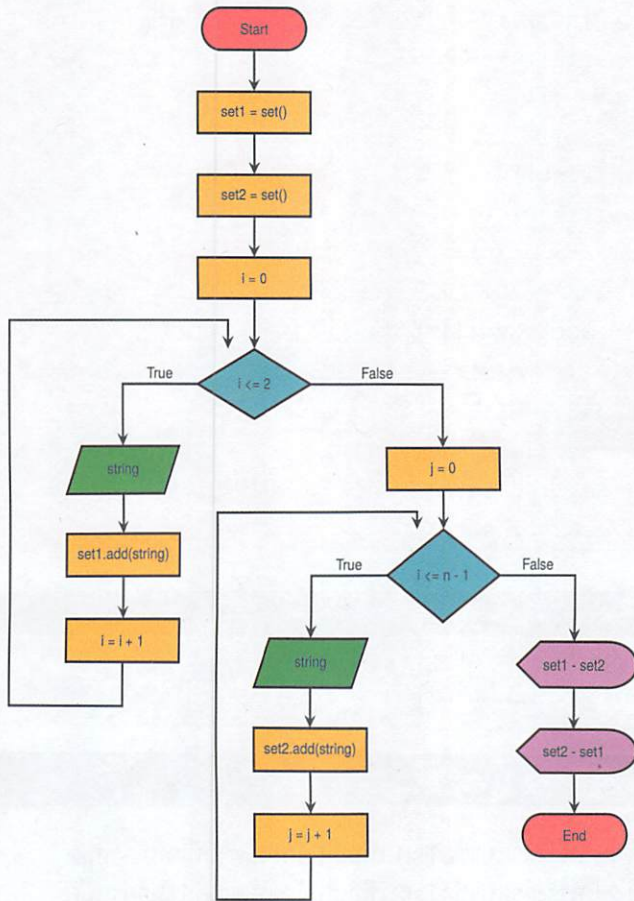
เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 เพื่อเก็บอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : สร้างตัวแปรชื่อ list2 เพื่อเก็บอักขระตามที่โจทย์กำหนด
บรรทัดที่ 3-4 : สร้างตัวแปรชื่อ set1 และ set2 และกำหนดให้เป็น empty set
บรรทัดที่ 5 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list1
บรรทัดที่ 6 : สร้างตัวแปรชื่อ m และกำหนดค่าเป็นความยาวของ list2
บรรทัดที่ 7 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 8-9 : ถ้า i เป็นเลขคู่ ให้เพิ่มสมาชิกตำแหน่งที่ i ของ list1 ใส่ใน set1
บรรทัดที่ 10 : พิจารณาค่า i ตั้งแต่ 0 ถึง m-1
บรรทัดที่ 11-12 : ถ้า i เป็นเลขคี่ ให้เพิ่มสมาชิกตำแหน่งที่ i ของ list2 ใส่ใน set2
บรรทัดที่ 13 : พิมพ์ค่า set1 ออกมา
บรรทัดที่ 14 : พิมพ์ค่า set2 ออกมา

เฉลยข้อที่ 85. แนวคิดเดียวกับข้อ 84

เฉลยข้อที่ 86. แนวคิดเดียวกับข้อ 84

เฉลยข้อที่ 87.



12

Input :

```
set1 = set()
set2 = set()
for i in range(3):
    string = input('Please insert element to set1: ')
    set1.add(string)
for j in range(3):
    string = input('Please insert element to set2: ')
    set2.add(string)
print('set1 - set2' : , set1 - set2)
print('set2 - set1' : , set2 - set1)
```

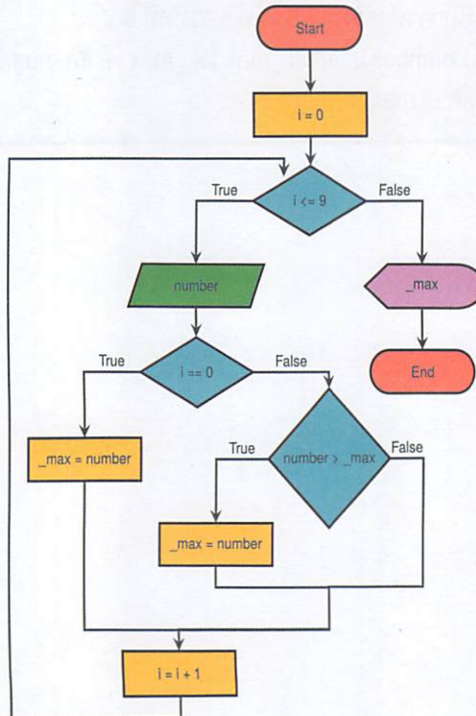
Output :

```
Please insert element to set1: a
Please insert element to set1: b
Please insert element to set1: c
Please insert element to set2: c
Please insert element to set2: d
Please insert element to set2: e
set1 - set2 : {'a', 'b'}
set2 - set1 : {'e', 'd'}
```

เฉลยละเอียด

บรรทัดที่ 1-2 : สร้างตัวแปรชื่อ list1 และกำหนดให้เป็น empty set
บรรทัดที่ 3 : วนทำบรรทัดที่ 4-5 ซ้ำ 3 ครั้ง
บรรทัดที่ 4-5 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ string แล้วนำไปเพิ่มใน set1
บรรทัดที่ 6 : วนทำบรรทัดที่ 7-8 ซ้ำ 3 ครั้ง
บรรทัดที่ 7-8 : รับอินพุตที่เป็นสายอักขระ และกำหนดค่าใส่ตัวแปรชื่อ string แล้วนำไปเพิ่มใน set2
บรรทัดที่ 9 : พิมพ์ค่า set1 - set2 ออกมา
บรรทัดที่ 10 : พิมพ์ค่า set2 - set1 ออกมา

เฉลยข้อที่ 88.



12

Input :

```
1 for i in range(5):
2     number = float(input('Please insert number ' + str(i+1) + ': ' ))
3     if i == 0:
4         _max = number
5     else:
6         if number > _max:
7             _max = number
8 print(_max)
```

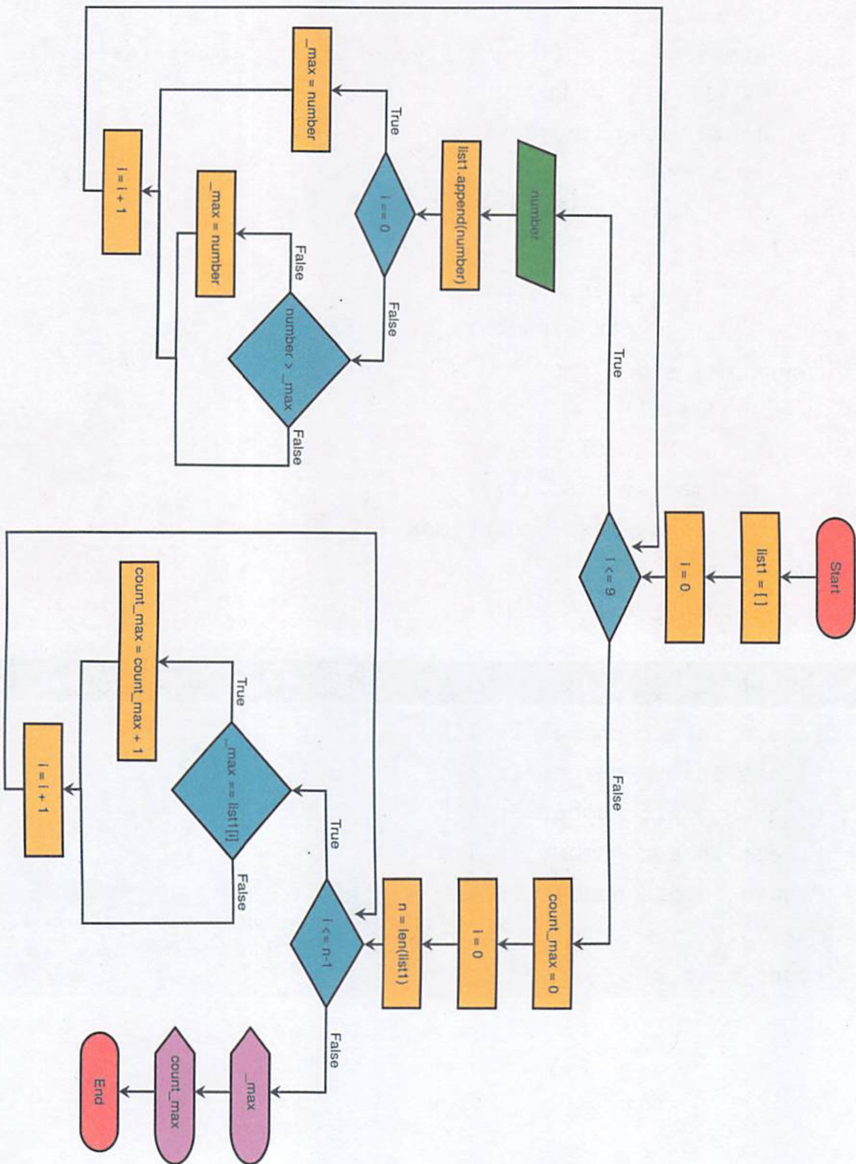
Output :

```
Please insert number 1: 1
Please insert number 2: 5
Please insert number 3: 2
Please insert number 4: 4
Please insert number 5: 3
5.0
```

เฉลยละเอียด

บรรทัดที่ 1 : วนทำบรรทัดที่ 2-7 ซ้ำ 5 ครั้ง
บรรทัดที่ 2 : รับอินพุตที่เป็นจำนวนจริง และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 3-4 : ถ้าเป็นการทำครั้งแรก (i=0) ให้ _max เป็น number
บรรทัดที่ 5 : ถ้าเป็นการทำครั้งที่ 2-5 ให้ทำบรรทัดที่ 6-7
บรรทัดที่ 6-7 : ถ้า number มากกว่า _max ให้ _max เท่ากับ number
บรรทัดที่ 8 : พิมพ์ค่า _max ออกมา

เฉลยข้อที่ 89.


Start
list1 = [ ]
i = 0
i <= 9
True
number
list1.append(number)
i == 0
True
_max = number
False
number > _max
False
_max = number
False
i = i + 1
False
count_max = 0
i = 0
n = len(list1)
i <= n-1
True
_max == list1[i]
True
count_max = count_max + 1
False
i = i + 1
False
_max
count_max
End

Input :

```
list1 = []
for i in range(5):
    number = float(input('Please insert number ' + str(i
    ↪ +1) + ': ' ))
    list1.append(number)
    if i == 0:
        _max = number
    else:
        if number > _max:
            _max = number
count_max = 0
n = len(list1)
for i in range(n):
    if _max == list1[i]:
        count_max = count_max + 1
print(_max)
print(count_max)
```

Output :

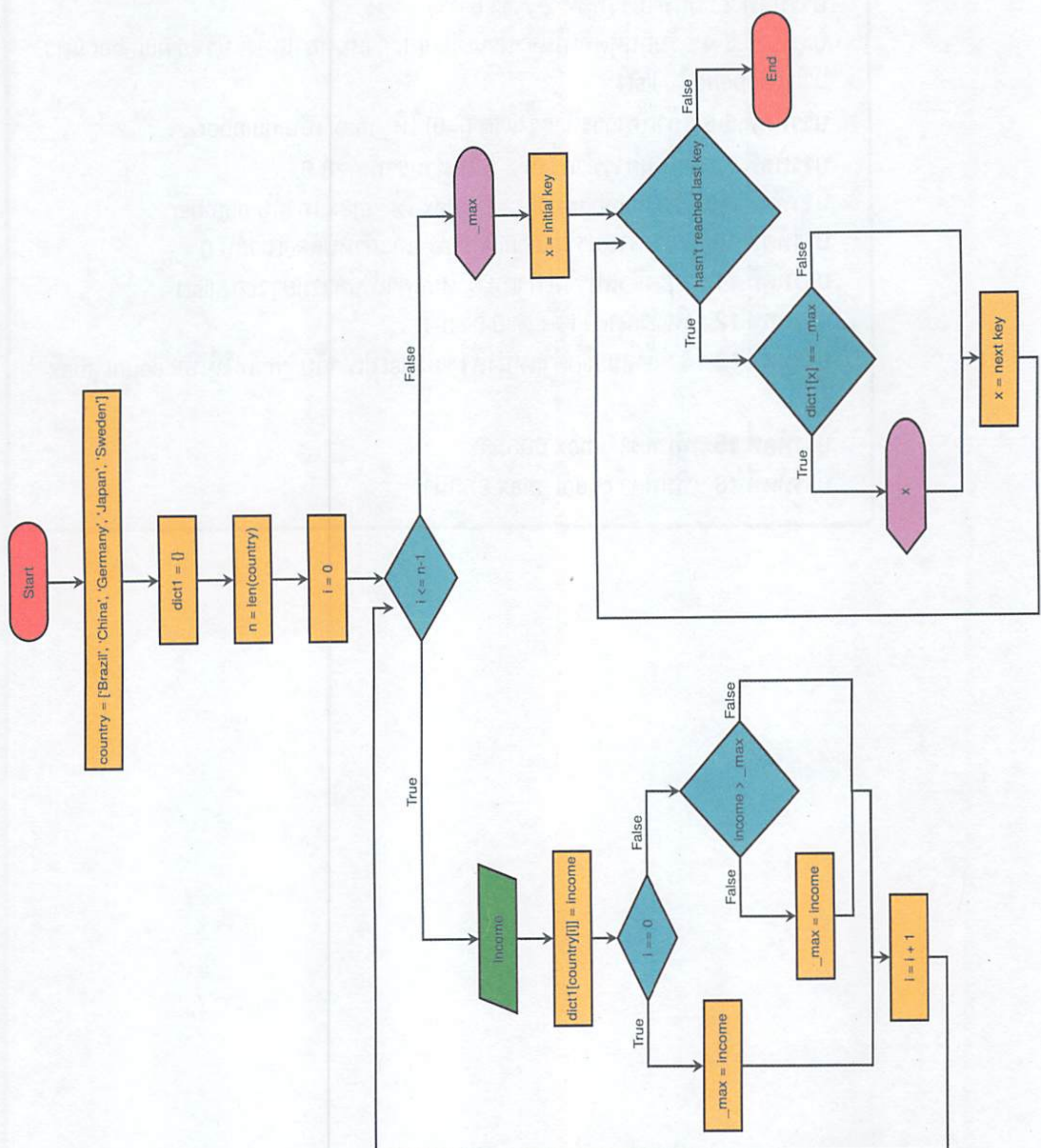
```
Please insert number 1: 1
Please insert number 2: 2
Please insert number 3: 5
Please insert number 4: 5
Please insert number 5: 4
max: 5.0
count_max: 2
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 และกำหนดให้เป็น empty list
บรรทัดที่ 2 : วนทำบรรทัดที่ 2-7 ซ้ำ 5 ครั้ง
บรรทัดที่ 3-4 : รับอินพุตที่เป็นจำนวนเต็ม และกำหนดค่าใส่ตัวแปรชื่อ number แล้วนำไป append ใน list1
บรรทัดที่ 5-6 : ถ้าเป็นการทำครั้งแรก (i=0) ให้ _max เป็น number
บรรทัดที่ 7 : ถ้าเป็นการทำครั้งที่ 2-5 ให้ทำบรรทัดที่ 8-9
บรรทัดที่ 8-9 : ถ้า number มากกว่า _max ให้ _max เท่ากับ number
บรรทัดที่ 10 : สร้างตัวแปรชื่อ count_max และกำหนดค่าเท่ากับ 0
บรรทัดที่ 11 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ list1
บรรทัดที่ 12 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 13-14 : ถ้าสมาชิกตำแหน่งที่ i ของ list1 เท่ากับ _max ให้เพิ่ม count_max ขึ้น 1
บรรทัดที่ 15 : พิมพ์ค่า _max ออกมา
บรรทัดที่ 16 : พิมพ์ค่า count_max ออกมา

12

เฉลยข้อที่ 90.


Start
country = ['Brazil', 'China', 'Germany', 'Japan', 'Sweden']
dict1 = {}
n = len(country)
i = 0
i <= n-1
True
False
income
dict1[country[i]] = income
i == 0
True
False
_max = income
income > _max
False
False
_max = income
i = i + 1
_max
x = initial key
hasn't reached last key
True
False
End
dict1[x] == _max
True
False
x
x = next key

Input :

```
country = ['Brazil', 'China', 'Germany', 'Japan', 'Sweden']
dict1 = {}
n = len(country)
for i in range(n):
    income = float(input('Please insert ' + country[i] + ' income: '))
    dict1[country[i]] = income
    if i == 0:
        _max = income
    else:
        if income > _max:
            _max = income
print('max income:', _max)
for x in dict1:
    if dict1[x] == _max:
        print('highest income country:', x)
```

Output :

```
Please insert Brazil income: 3500
Please insert China income: 2500
Please insert Germany income: 2000
Please insert Japan income: 3000
Please insert Sweden income: 1500
max income: 3500.0
highest income country: Brazil
```

เฉลยละเอียด

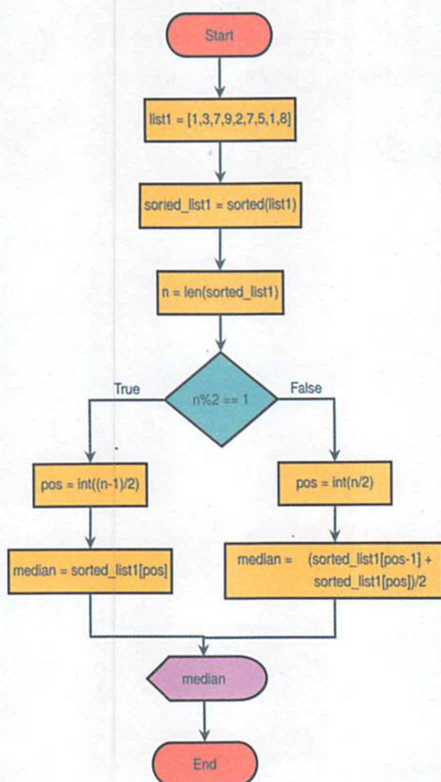
บรรทัดที่ 1 : สร้างตัวแปรชื่อ country เพื่อเก็บสายอักขระตามที่โจทย์กำหนด
บรรทัดที่ 2 : สร้างตัวแปรชื่อ dict1 และกำหนดให้เป็น empty dictionary
บรรทัดที่ 3 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ country
บรรทัดที่ 4 : พิจารณาค่า i ตั้งแต่ 1 ถึง n
บรรทัดที่ 5 : รับอินพุตที่เป็นจำนวนจริง และกำหนดค่าใส่ตัวแปรชื่อ income
บรรทัดที่ 6 : กำหนดให้ dict1 มี key เป็นสมาชิกตำแหน่งที่ i ของ country และมี value เป็น income
บรรทัดที่ 7-8 : ถ้าค่า i เท่ากับ 0 ให้ _max เท่ากับ income
บรรทัดที่ 9 : ถ้า i ไม่เท่ากับ 0 ให้ทำบรรทัดที่ 10-11
บรรทัดที่ 10-11 : ถ้า income มากกว่า _max ให้ _max เท่ากับ income
บรรทัดที่ 12 : พิมพ์ค่า _max ออกมา
บรรทัดที่ 13 : พิจารณาแต่ละ key ของ dict1
บรรทัดที่ 14-15 : ถ้า value ของ key เท่ากับค่า _max ให้พิมพ์ key ออกมา

เฉลยข้อที่ 91. แนวคิดเดียวกับข้อ 88

เฉลยข้อที่ 92. แนวคิดเดียวกับข้อ 89

เฉลยข้อที่ 93. แนวคิดเดียวกับข้อ 90

เฉลยข้อที่ 94.



12

Input :

```
1 list1 = [1, 3, 7, 9, 2, 7, 5, 1, 8]
2 sorted_list1 = sorted(list1)
3 n = len(sorted_list1)
4 if n%2 == 1:
5     pos = int((n-1)/2)
6     median = sorted_list1[pos]
7 else:
8     pos = int(n/2)
9     median = (sorted_list1[pos-1] + sorted_list1[pos])/2
10 print(median)
```

Output :

```
5
```

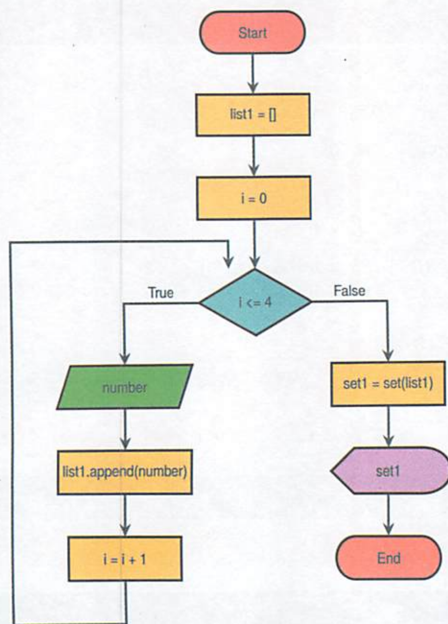
เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 เพื่อเก็บเลขตามที่โจทย์กำหนด
บรรทัดที่ 2 : สร้างตัวแปรชื่อ sorted_list1 เพื่อเรียง list1 จากน้อยไปมาก
บรรทัดที่ 3 : สร้างตัวแปรชื่อ n และกำหนดค่าเป็นความยาวของ sorted_list1
บรรทัดที่ 4 : ถ้า n เป็นเลขคี่ ให้ทำบรรทัดที่ 5-6
บรรทัดที่ 5 : สร้างตัวแปรชื่อ pos เพื่อคำนวณค่ากลางของ n ในรูปของจำนวนเต็ม
บรรทัดที่ 6 : สร้างตัวแปรชื่อ median และกำหนดค่าเท่ากับสมาชิกของ sorted_list1 ตำแหน่งที่ pos
บรรทัดที่ 7 : ถ้า n เป็นเลขคู่ ให้ทำบรรทัดที่ 8-9
บรรทัดที่ 8 : สร้างตัวแปรชื่อ pos เพื่อคำนวณค่ากลางของ n ในรูปแบบของจำนวนเต็ม
บรรทัดที่ 9 : สร้างตัวแปรชื่อ median และกำหนดค่าเท่ากับ ค่าเฉลี่ยของสมาชิกของ sorted_list1 ในตำแหน่งที่ pos-1 และ pos
บรรทัดที่ 10 : พิมพ์ค่า median ออกมา

เฉลยข้อที่ 95. แนวคิดเดียวกับข้อ 94

เฉลยข้อที่ 96. แนวคิดเดียวกับข้อ 94

เฉลยข้อที่ 97.



Input :

```
1 list1 = []
2 for i in range(5):
3     number = float(input('Please insert number ' + str(i
    ↪ +1) + ': ' ))
4     list1.append(number)
5 set1 = set(list1)
6 print(set1)
```

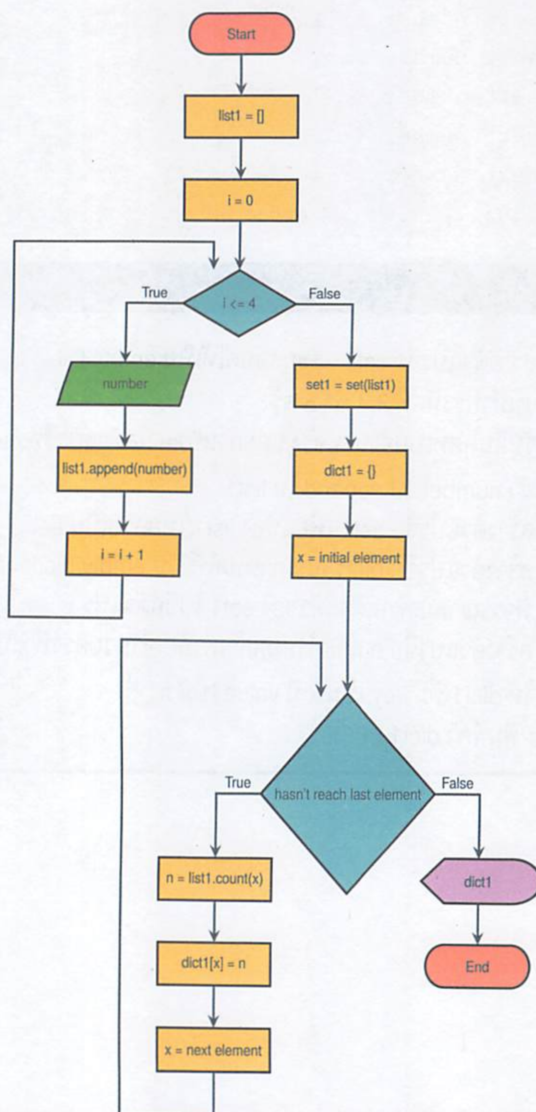
Output :

```
Please insert number 1: 5
Please insert number 2: 3
Please insert number 3: 2
Please insert number 4: 3
Please insert number 5: 4
{2.0, 3.0, 4.0, 5.0}
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 และกำหนดให้เป็น empty list
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-4 ซ้ำ 5 ครั้ง
บรรทัดที่ 3 : รับอินพุตที่เป็นจำนวนจริง และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 4 : นำ number ไป append ใน list1
บรรทัดที่ 5 : สร้างตัวแปรชื่อ set1 เพื่อแปลง list1 ให้กลายเป็น set()
บรรทัดที่ 6 : พิมพ์ค่า set1 ออกมา

เฉลยข้อที่ 98.



12

Input :

```
list1 = []
for i in range(5):
    number = float(input('Please insert number ' + str(i
    ↪ +1) + ': ' ))
    list1.append(number)
set1 = set(list1)
dict1 = {}
for x in set1:
    n = list1.count(x)
    dict1[x] = n
print(dict1)
```

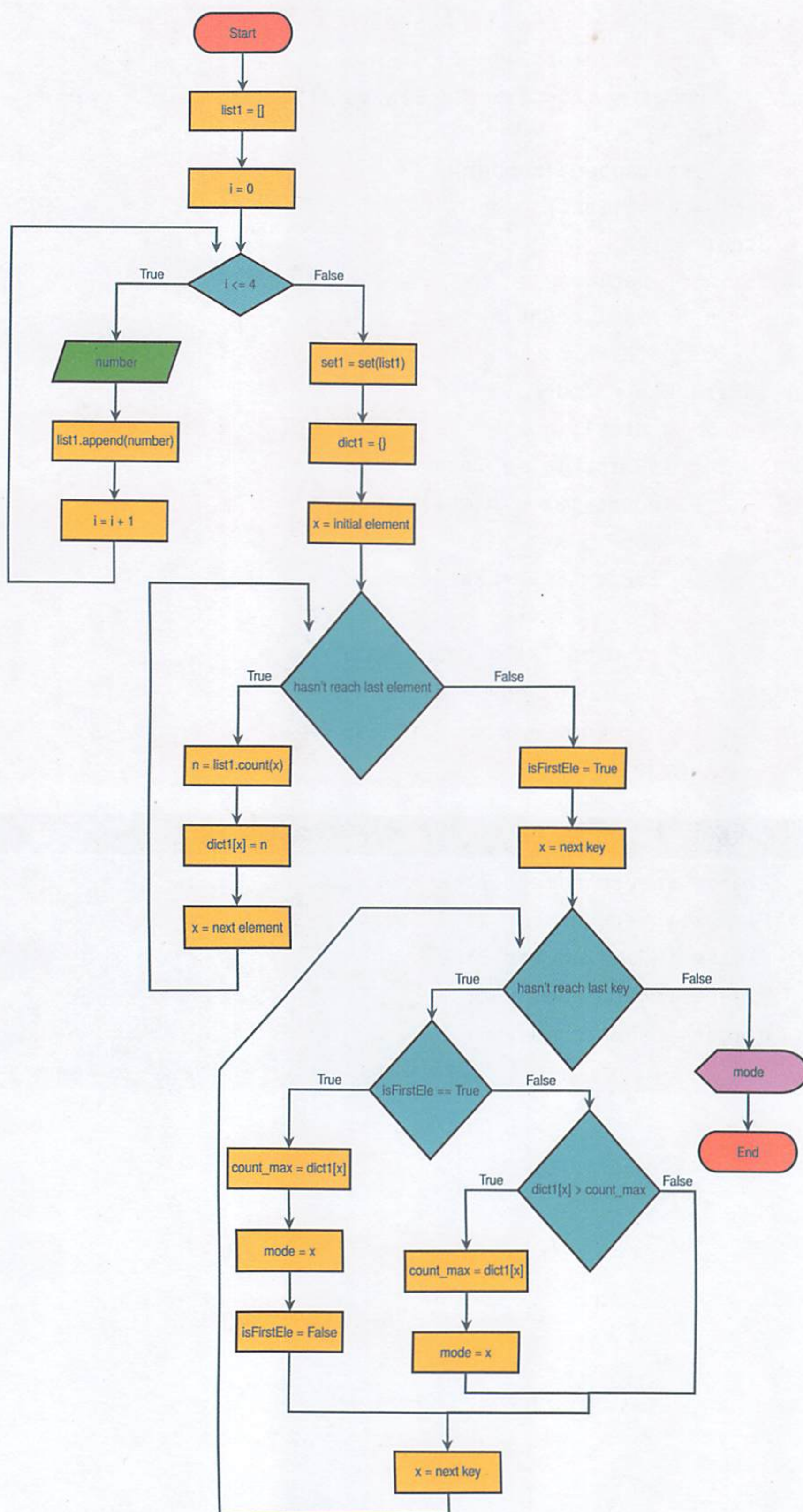
Output :

```
Please insert number 1: 1
Please insert number 2: 3
Please insert number 3: 3
Please insert number 4: 2
Please insert number 5: 4
{1.0: 1, 2.0: 1, 3.0: 2, 4.0: 1}
```

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 และกำหนดให้เป็น empty list
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-4 ซ้ำ 5 ครั้ง
บรรทัดที่ 3 : รับอินพุตที่เป็นจำนวนจริง และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 4 : นำ number ไป append ใน list1
บรรทัดที่ 5 : สร้างตัวแปรชื่อ set1 เพื่อแปลง list1 ให้กลายเป็น set
บรรทัดที่ 6 : สร้างตัวแปรชื่อ dict1 และกำหนดให้เป็น empty dictionary
บรรทัดที่ 7 : พิจารณาสมาชิกแต่ละตัวของ set1 ในชื่อตัวแปร x
บรรทัดที่ 8 : สร้างตัวแปรชื่อ n และกำหนดค่าเท่ากับ จำนวนครั้งที่ x ปรากฏใน list1
บรรทัดที่ 9 : ให้ dict1 ที่มี key เป็น x มี value เป็น n
บรรทัดที่ 10 : พิมพ์ค่า dict1 ออกมา

เฉลยข้อที่ 99.


Start
list1 = []
i = 0
i <= 4
True
False
number
list1.append(number)
i = i + 1
set1 = set(list1)
dict1 = {}
x = initial element
hasn't reach last element
True
False
n = list1.count(x)
dict1[x] = n
x = next element
isFirstEle = True
x = next key
hasn't reach last key
True
False
mode
End
isFirstEle == True
True
False
count_max = dict1[x]
mode = x
isFirstEle = False
dict1[x] > count_max
True
False
count_max = dict1[x]
mode = x
x = next key

Input :

```
list1 = []
for i in range(5):
    number = float(input('Please insert number ' + str(i
    ↪ +1) + ': ' ))
    list1.append(number)
set1 = set(list1)
dict1 = {}
for x in set1:
    n = list1.count(x)
    dict1[x] = n
isFirstEle = True
for x in dict1:
    if isFirstEle == True:
        count_max = dict1[x]
        mode = x
        isFirstEle = False
    else:
        if dict1[x] > count_max:
            count_max = dict1[x]
            mode = x
print(mode)
```

Output :

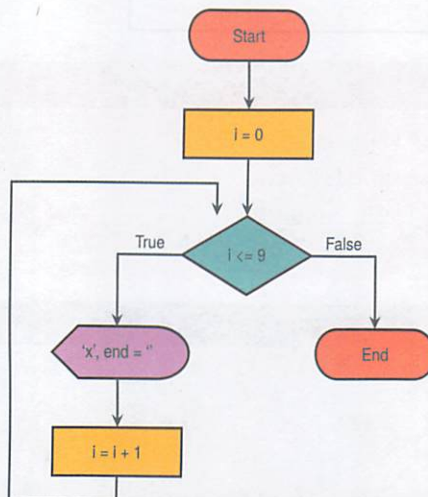
```
Please insert number 1: 2
Please insert number 2: 3
Please insert number 3: 4
Please insert number 4: 3
Please insert number 5: 3
3.0
```

12

เฉลยละเอียด

บรรทัดที่ 1 : สร้างตัวแปรชื่อ list1 และกำหนดให้เป็น empty list
บรรทัดที่ 2 : วนทำบรรทัดที่ 3-4 ซ้ำ 5 ครั้ง
บรรทัดที่ 3 : รับอินพุตที่เป็นจำนวนจริง และกำหนดค่าใส่ตัวแปรชื่อ number
บรรทัดที่ 4 : นำ number ไป append ใน list1
บรรทัดที่ 5 : สร้างตัวแปรชื่อ set1 เพื่อแปลง list1 ให้กลายเป็น set
บรรทัดที่ 6 : สร้างตัวแปรชื่อ dict1 และกำหนดให้เป็นempty dictionary
บรรทัดที่ 7 : พิจารณาสมาชิกแต่ละตัวของ set1 ในชื่อตัวแปร x
บรรทัดที่ 8 : สร้างตัวแปรชื่อ n และกำหนดค่าเท่ากับ จำนวนครั้งที่ x ปรากฏใน list1
บรรทัดที่ 9 : ให้ dict1 ที่มี key เป็น x มี value เป็น n
บรรทัดที่ 10 : สร้างตัวแปรชื่อ isFirstEle และกำหนดค่าเท่ากับ True
บรรทัดที่ 11 : พิจารณา key แต่ละค่าใน dict1
บรรทัดที่ 12 : ถ้า isFirstEle เป็น True ให้ทำบรรทัดที่ 13-15
บรรทัดที่ 13 : สร้างตัวแปรชื่อ count_max เพื่อเก็บ value ของ dict1 ที่มี key เป็น x
บรรทัดที่ 14 : สร้างตัวแปรชื่อ mode และกำหนดค่าเท่ากับ x
บรรทัดที่ 15 : ให้ isFirstEle เป็น False
บรรทัดที่ 16 : ถ้า isFirstEle เป็น False
บรรทัดที่ 17 : ถ้า value ของ x ใน dict1 มากกว่า count_max
บรรทัดที่ 18 - 19 : สร้างตัวแปรและเก็บค่าเหมือนบรรทัดที่ 13-14
บรรทัดที่ 20 : พิมพ์ค่า mode ออกมา

เฉลยข้อที่ 100.



12

Input :

```
for i in range(10):
    print('x', end='')
```

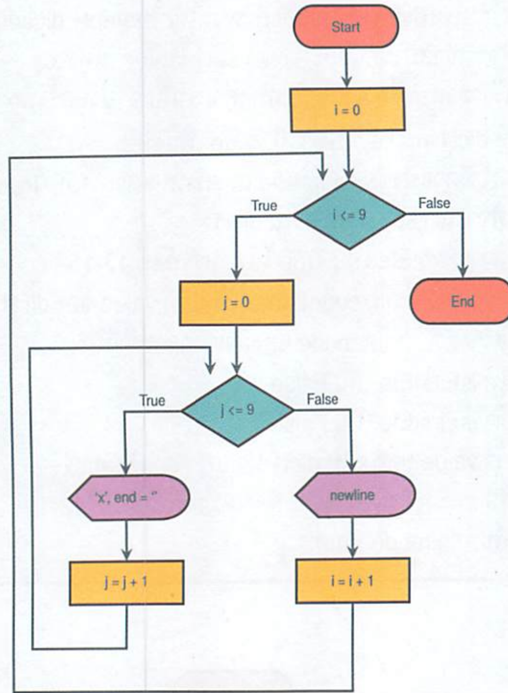
Output :

```
xxxxxxxxxx
```

เฉลยละเอียด

บรรทัดที่ 1 : วนทำบรรทัดที่ 2 ซ้ำ 10 ครั้ง
บรรทัดที่ 2 : พิมพ์ 'x' โดยที่ไม่ต้องขึ้นบรรทัดใหม่

เฉลยข้อที่ 101.



12

Input :

```
for i in range(10):
    for j in range(10):
        print('x', end='')
    print()
```

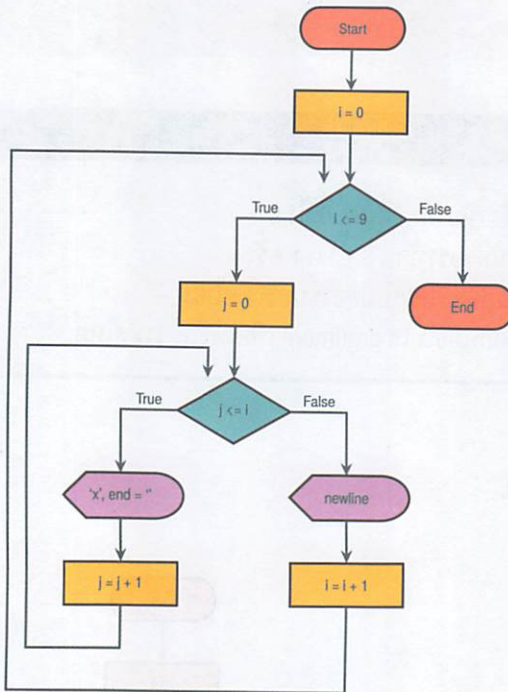
Output :

```
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
xxxxxxxxxx
```

เฉลยละเอียด

บรรทัดที่ 1 : วนทำบรรทัดที่ 2-4 ซ้ำ 10 ครั้ง
บรรทัดที่ 2 : วนทำบรรทัดที่ 3 ซ้ำ 10 ครั้ง
บรรทัดที่ 3 : พิมพ์ 'x' โดยที่ไม่ต้องขึ้นบรรทัดใหม่
บรรทัดที่ 4 : พิมพ์โดยไม่ใส่ argument เพื่อให้ขึ้นบรรทัดใหม่

เฉลยข้อที่ 102.



Input :

```
for i in range(10):
    for j in range(i+1):
        print('x', end='')
    print()
```

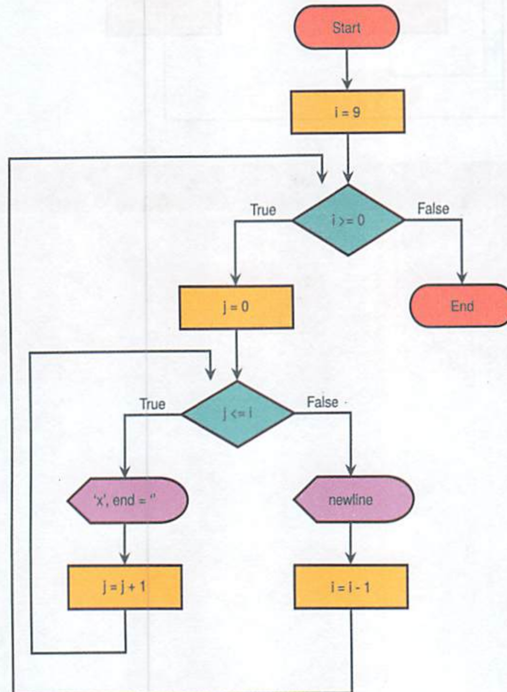
Output :

```
x
xx
xxx
xxxx
xxxxx
xxxxxx
xxxxxxx
xxxxxxxx
xxxxxxxxx
xxxxxxxxxx
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 0 ถึง 9
บรรทัดที่ 2 : วนทำบรรทัดที่ 3 ซ้ำ i+1 ครั้ง
บรรทัดที่ 3 : พิมพ์ 'x' โดยที่ไม่ต้องขึ้นบรรทัดใหม่
บรรทัดที่ 4 : พิมพ์โดยไม่ใส่ argument เพื่อให้ขึ้นบรรทัดใหม่

เฉลยข้อที่ 103.



12

Input :

```
1 for i in range(9, -1, -1):
2     for j in range(i+1):
3         print('x', end='')
4     print()
```

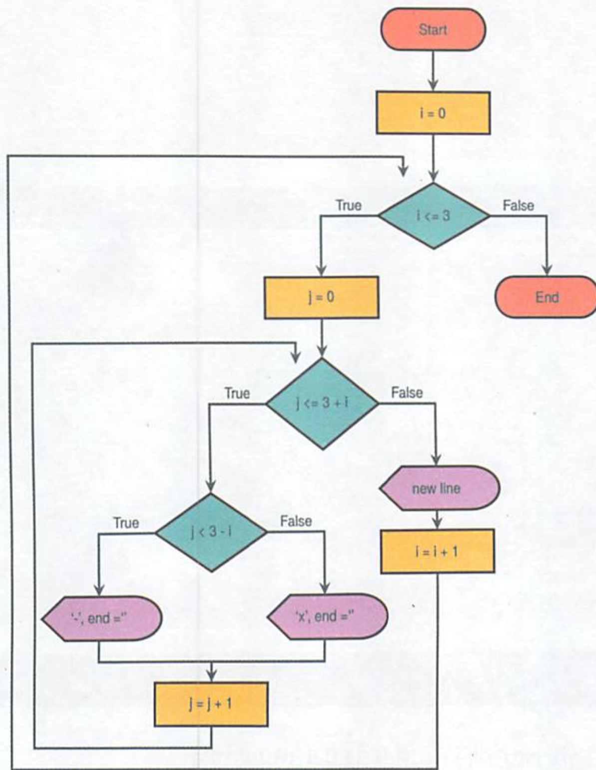
Output :

```
xxxxxxxxxx
xxxxxxxxx
xxxxxxxx
xxxxxxx
xxxxxx
xxxxx
xxxx
xxx
xx
x
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 9 ถึง 0 แบบถอยหลัง
บรรทัดที่ 2 : วนทำบรรทัดที่ 3 ซ้ำ i+1 ครั้ง
บรรทัดที่ 3 : พิมพ์ 'x' โดยที่ไม่ต้องขึ้นบรรทัดใหม่
บรรทัดที่ 4 : พิมพ์โดยไม่ใส่ argument เพื่อให้ขึ้นบรรทัดใหม่

เฉลยข้อที่ 104.



Input :

```
for i in range(4):
    for j in range(4+i):
        if j < 3-i:
            print('-', end='')
        else:
            print('x', end='')
    print()
```

Output :

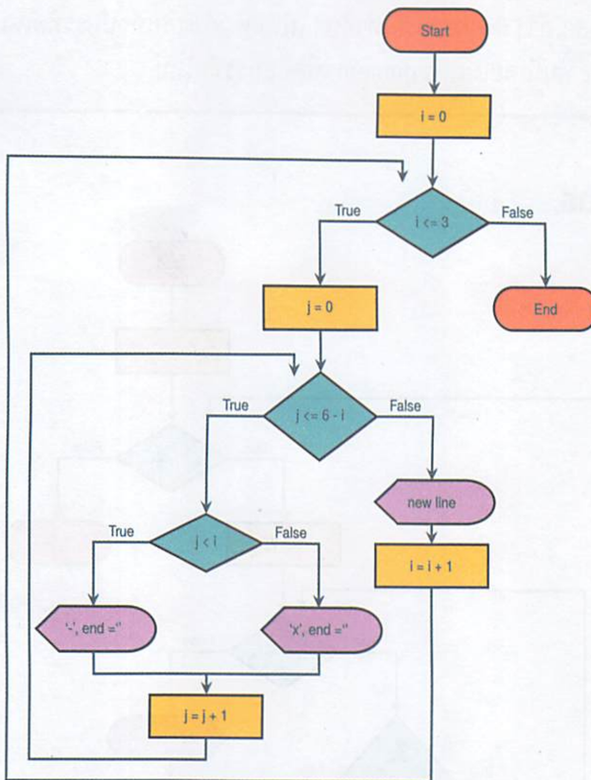
```
---x
--xxx
-xxxxx
xxxxxxx
```

12

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 0 ถึง 3
บรรทัดที่ 2 : พิจารณาค่า j ตั้งแต่ 0 ถึง i+3
บรรทัดที่ 3-4 : ถ้า j น้อยกว่า 3-i ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 5-6 : ถ้า j มากกว่าหรือเท่ากับ 3-i ให้พิมพ์ 'x' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 7 : พิมพ์โดยไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

เฉลยข้อที่ 105.



Input :

```
1 for i in range(4):
2     for j in range(7-i):
3         if j < i:
4             print('-', end='')
5         else:
6             print('x', end='')
7     print()
```

12

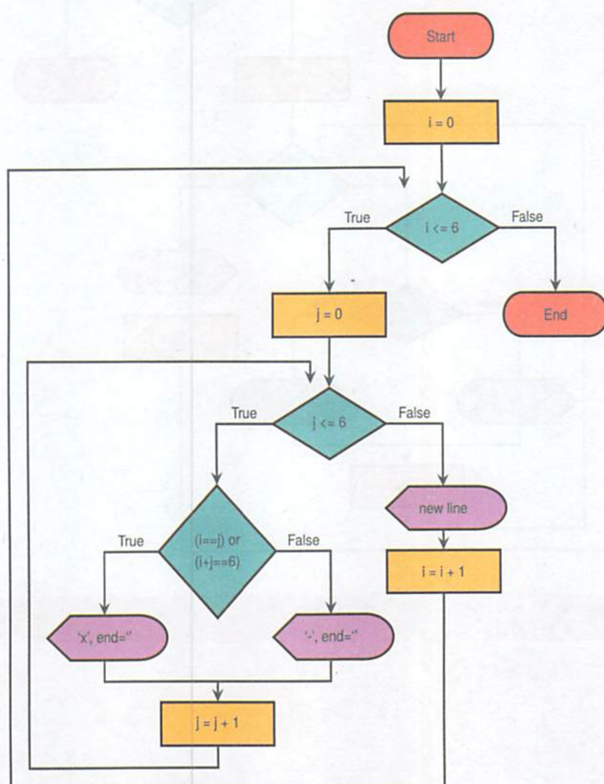
Output :

```
xxxxxxx
-xxxxx
--xxx
---x
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 0 ถึง 3
บรรทัดที่ 2 : พิจารณาค่า j ตั้งแต่ 0 ถึง 7-i
บรรทัดที่ 3-4 : ถ้า j น้อยกว่า i ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 5-6 : ถ้า j มากกว่าหรือเท่ากับ i ให้พิมพ์ 'x' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 7 : พิมพ์โดยไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

เฉลยข้อที่ 106.



Input :

```
1 for i in range(7):
2     for j in range(7):
3         if (i==j) or (i+j == 6):
4             print('x', end='')
5         else:
6             print('-', end='')
7     print()
```

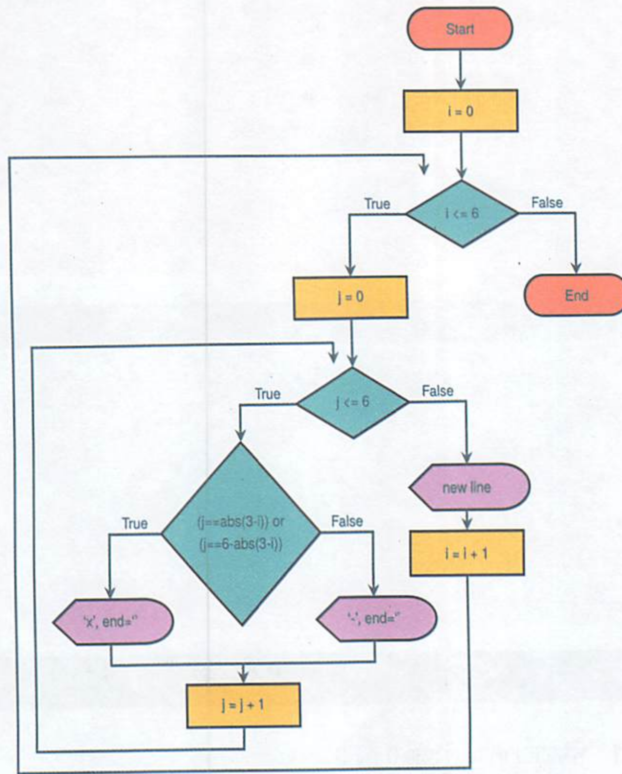
Output :

```
x-----x
-x---x-
--x-x--
---x---
--x-x--
-x---x-
x-----x
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 0 ถึง 6
บรรทัดที่ 2 : พิจารณาค่า j ตั้งแต่ 0 ถึง 6
บรรทัดที่ 3-4 : ถ้า i เท่ากับ j หรือ i+j เท่ากับ 6 ให้พิมพ์ 'x' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 5-6 : ถ้าไม่ใช่เงื่อนไขตามบรรทัดที่ 3 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 7 : พิมพ์โดยไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

เฉลยข้อที่ 107.



12

Input :

```
for i in range(7):
    for j in range(7):
        if (j==abs(3-i)) or (j==6-abs(3-i)):
            print('x', end='')
        else:
            print('-', end='')
    print()
```

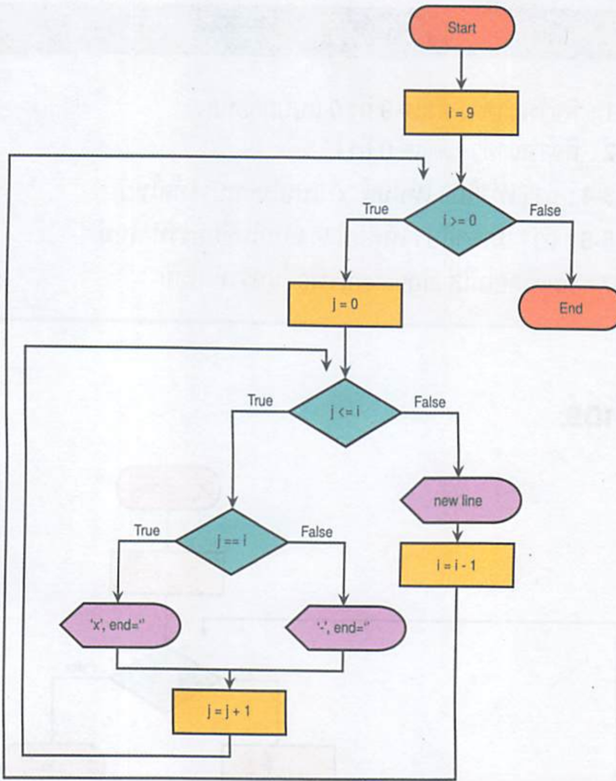
Output :

```
---x---
--x-x--
-x---x-
x-----x
-x---x-
--x-x--
---x---
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 0 ถึง 6
บรรทัดที่ 2 : พิจารณาค่า j ตั้งแต่ 0 ถึง 6
บรรทัดที่ 3-4 : ถ้า j เท่ากับ abs(3-i) หรือ j เท่ากับ 6-abs(3-i) ให้พิมพ์ 'x' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 5-6 : ถ้าไม่ใช่เงื่อนไขตามบรรทัดที่ 3 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 7 : พิมพ์โดยไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

เฉลยข้อที่ 108.



Input :

```python
for i in range(9, -1, -1):
    for j in range(i+1):
        if j == i:
            print('x', end='')
        else:
            print('-', end='')
    print()
```

Output :

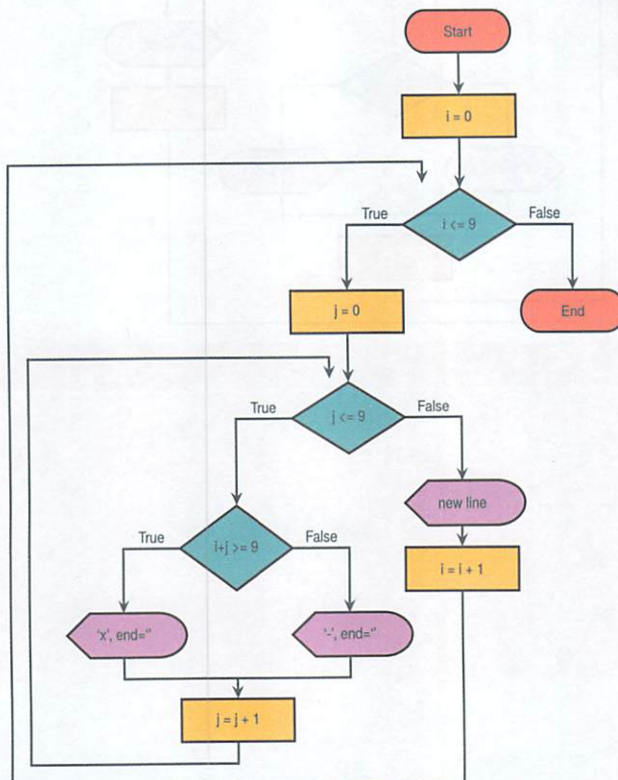
```
---------x
--------x
-------x
------x
-----x
----x
---x
--x
-x
x
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 9 ถึง 0 แบบถอยหลัง
บรรทัดที่ 2 : พิจารณาค่า j ตั้งแต่ 0 ถึง i
บรรทัดที่ 3-4 : ถ้า j เท่ากับ i ให้พิมพ์ 'x' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 5-6 : ถ้า j ไม่เท่ากับ i ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 7 : พิมพ์โดยไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

เฉลยข้อที่ 109.

12



Input :

```
1 for i in range(10):
2     for j in range(10):
3         if i+j >= 9:
4             print('x', end='')
5         else:
6             print('-', end='')
7     print()
```

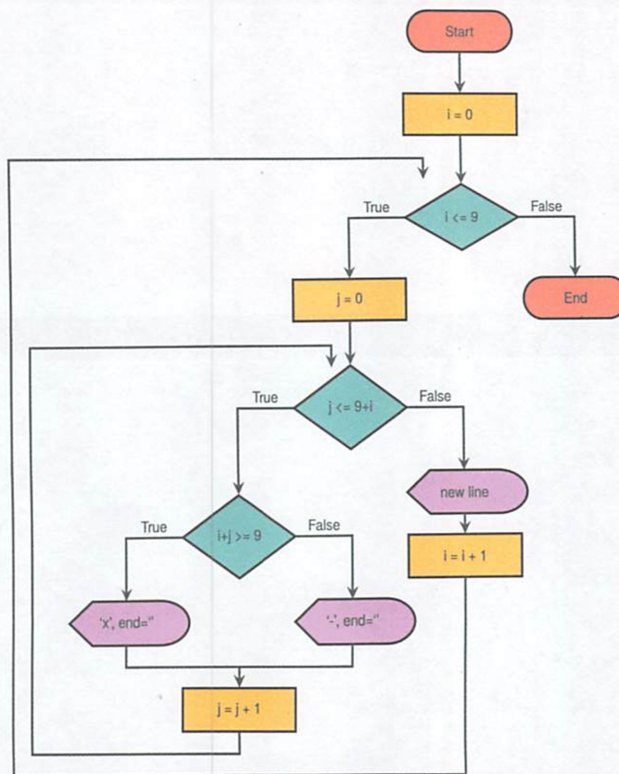
Output :

```
---------x
--------xx
-------xxx
------xxxx
-----xxxxx
----xxxxxx
---xxxxxxx
--xxxxxxxx
-xxxxxxxxx
xxxxxxxxxx
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 0 ถึง 9
บรรทัดที่ 2 : พิจารณาค่า j ตั้งแต่ 0 ถึง 9
บรรทัดที่ 3-4 : ถ้า i+j มากกว่าหรือเท่ากับ 9 ให้พิมพ์ 'x' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 5-6 : ถ้า i+j น้อยกว่า 9 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 7 : พิมพ์โดยไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

เฉลยข้อที่ 110.



Input :

```
for i in range(10):
    for j in range(10 + i):
        if i+j >= 9:
            print('x', end='')
        else:
            print('-', end='')
    print()
```

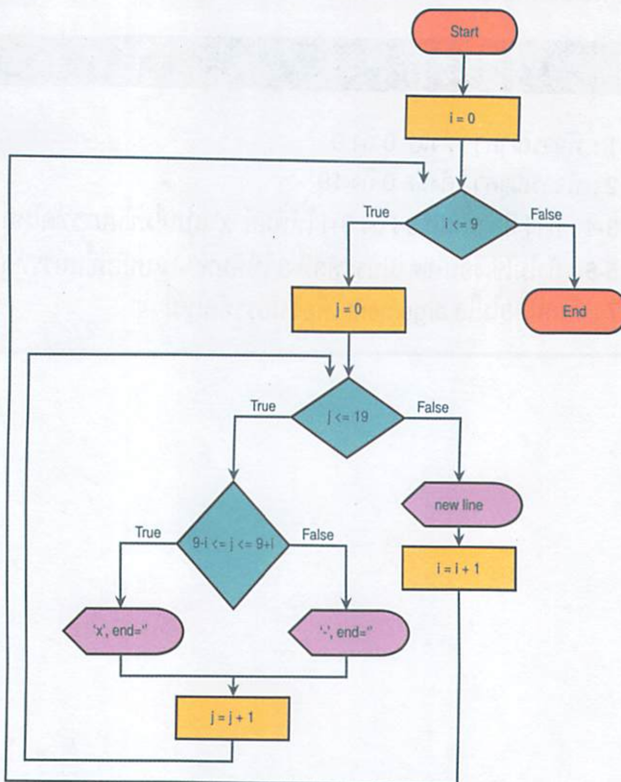
Output :

```
---------x
--------xxx
-------xxxxx
------xxxxxxx
-----xxxxxxxxx
----xxxxxxxxxxx
---xxxxxxxxxxxxx
--xxxxxxxxxxxxxxx
-xxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxx
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 0 ถึง 9
บรรทัดที่ 2 : พิจารณาค่า j ตั้งแต่ 0 ถึง i+9
บรรทัดที่ 3-4 : ถ้า i+j มากกว่าหรือเท่ากับ 9 ให้พิมพ์ 'x' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 5-6 : ถ้า i+j น้อยกว่า 9 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 7 : พิมพ์โดยไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

เฉลยข้อที่ 111.



Input :

```
for i in range(10):
    for j in range(20):
        if 9-i <= j <= 9+i:
            print('x', end='')
        else:
            print('-', end='')
    print()
```

Output :

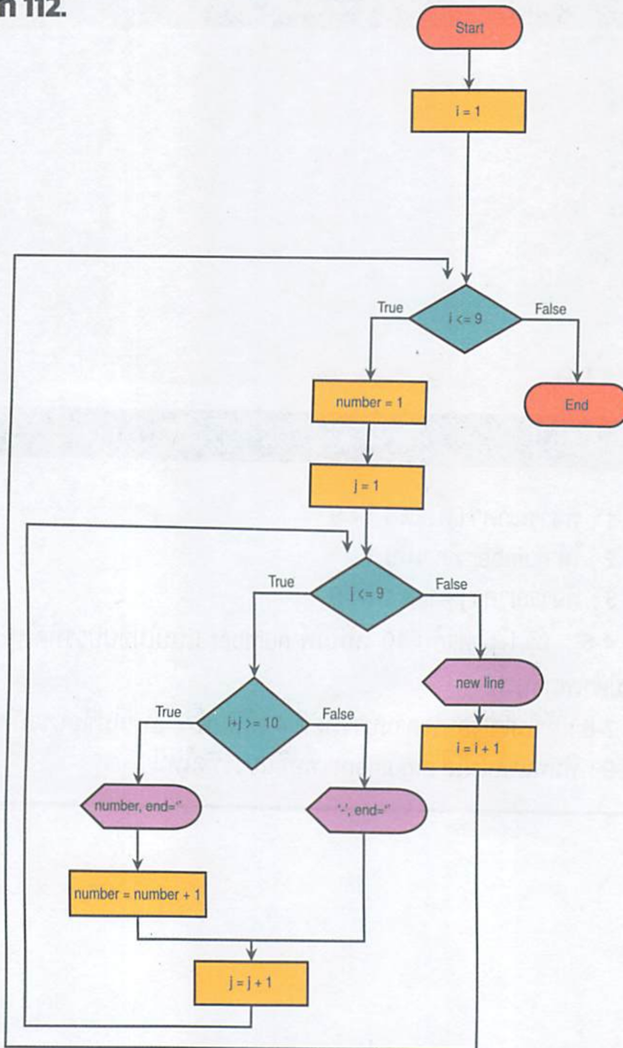
```
---------x----------
--------xxx---------
-------xxxxx--------
------xxxxxxx-------
-----xxxxxxxxx------
----xxxxxxxxxxx-----
---xxxxxxxxxxxxx----
--xxxxxxxxxxxxxxx---
-xxxxxxxxxxxxxxxxx--
xxxxxxxxxxxxxxxxxxx-
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 0 ถึง 9
บรรทัดที่ 2 : พิจารณาค่า j ตั้งแต่ 0 ถึง 19
บรรทัดที่ 3-4 : ถ้า j มีค่าตั้งแต่ 9-i ถึง 9+i ให้พิมพ์ 'x' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 5-6 : ถ้าไม่ใช่เงื่อนไขตามบรรทัดที่ 3 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 7 : พิมพ์โดยไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

12

เฉลยข้อที่ 112.



Input :

```
for i in range(1, 10):
    number = 1
    for j in range(1, 10):
        if i+j >= 10:
            print(number, end='')
            number = number + 1
        else:
            print('-', end='')
    print()
```

Output :

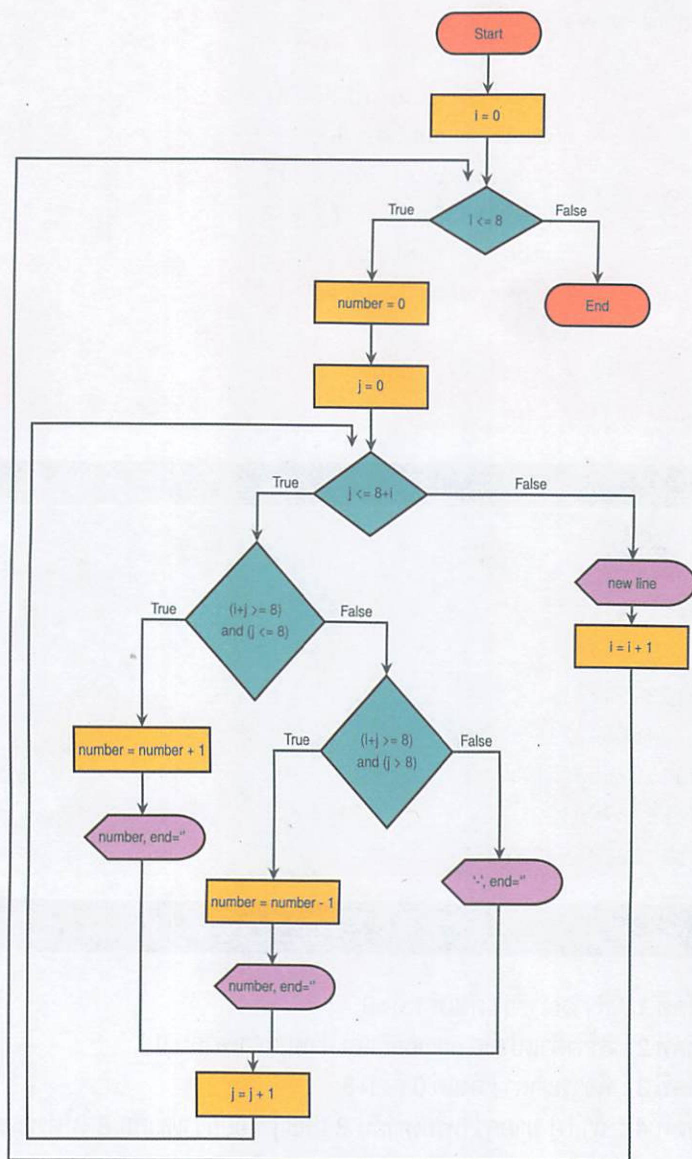
```
--------1
-------12
------123
-----1234
----12345
---123456
--1234567
-12345678
123456789
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 1 ถึง 9
บรรทัดที่ 2 : ให้ number มีค่าเป็น 1
บรรทัดที่ 3 : พิจารณาค่า j ตั้งแต่ 1 ถึง 9
บรรทัดที่ 4-6 : ถ้า i+j เท่ากับ 10 ให้พิมพ์ number แบบไม่ขึ้นบรรทัดใหม่ และให้ number มีค่าเพิ่มขึ้น 1
บรรทัดที่ 7-8 : ถ้าไม่ใช่เงื่อนไขตามบรรทัดที่ 4 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 9 : พิมพ์แบบไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

12

เฉลยข้อที่ 113.


Start
i = 0
True
i <= 8
False
End
number = 0
j = 0
True
j <= 8+i
False
True
(i+j >= 8)
and (j <= 8)
False
new line
i = i + 1
number = number + 1
True
(i+j >= 8)
and (j > 8)
False
number, end=''
'-', end=''
number = number - 1
number, end=''
j = j + 1

Input :

```
1 for i in range(9):
2     number = 0
3     for j in range(9+i):
4         if (i+j >= 8) and (j <= 8):
5             number = number + 1
6             print(number, end='')
7         elif (i+j >= 8) and (j > 8):
8             number = number - 1
9             print(number, end='')
10        else:
11            print('-', end='')
12    print()
```

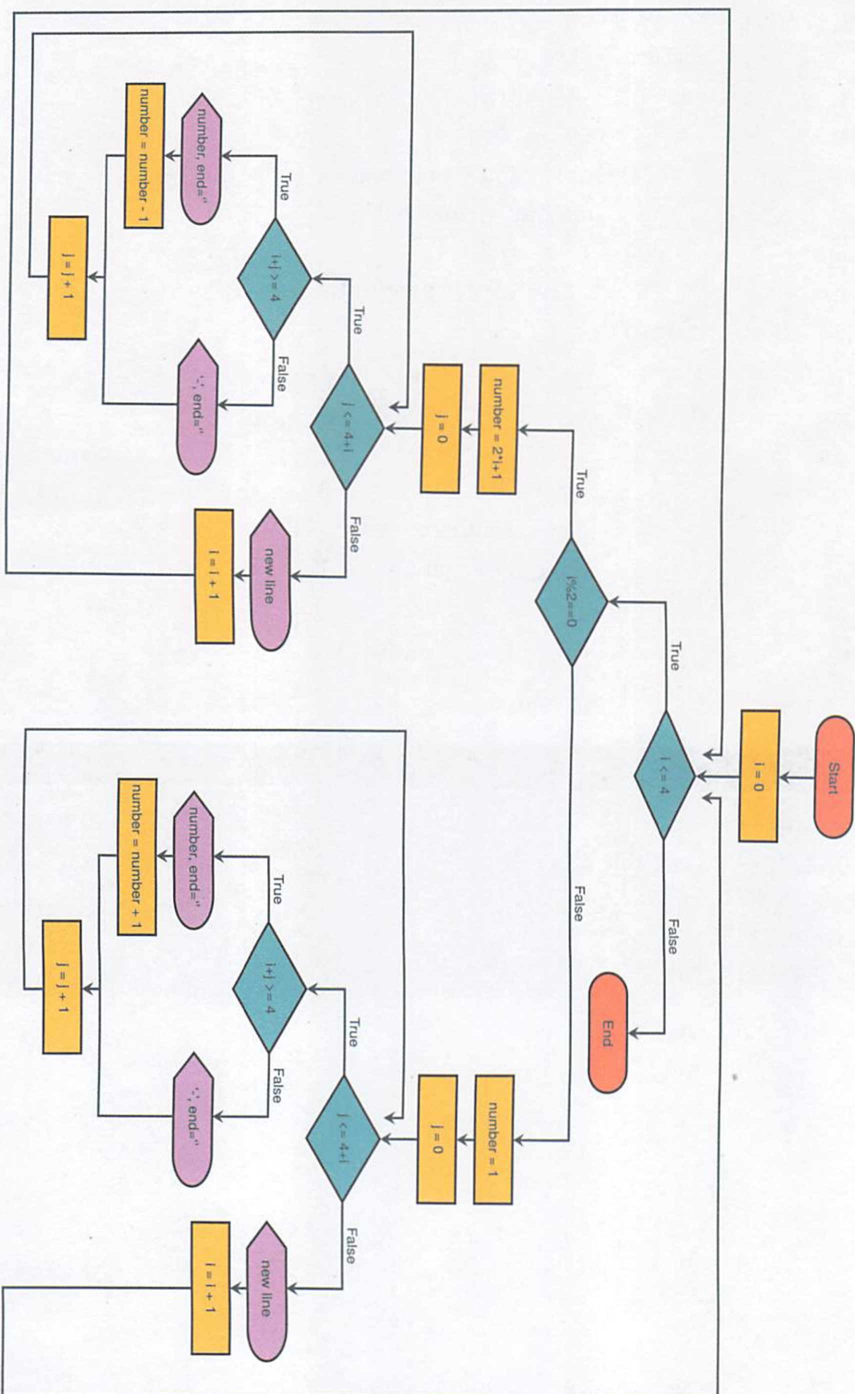
Output :

```
--------1
-------121
------12321
-----1234321
----123454321
---12345654321
--1234567654321
-123456787654321
12345678987654321
```


เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 1 ถึง 9
บรรทัดที่ 2 : สร้างตัวแปรชื่อ number และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3 : พิจารณาค่า j ตั้งแต่ 0 ถึง i+8
บรรทัดที่ 4 : ถ้า i+j มากกว่าหรือเท่ากับ 8 และ j น้อยกว่าเท่ากับ 8 ให้ทำบรรทัดที่ 5-6
บรรทัดที่ 5-6 : ให้ number มีค่าเพิ่มขึ้น 1 และ ให้พิมพ์ number แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 7 : ถ้า i+j มากกว่าหรือเท่ากับ 8 และ j มากกว่า 8 ให้ทำบรรทัดที่ 8-9
บรรทัดที่ 8-9 : ให้ number มีค่าลดลง 1 และ ให้พิมพ์ number แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 10-11 : ถ้าไม่ใช่เงื่อนไขตามบรรทัดที่ 4 และ 7 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 12 : พิมพ์แบบไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

เฉลยข้อที่ 114.


Start
i = 0
i <= 4
True
False
End
i%2==0
number = 2*i+1
j = 0
j <= 4+i
i+j >= 4
number, end=''
number = number - 1
j = j + 1
' ', end=''
new line
i = i + 1
number = 1
number = number + 1

Input :

```
for i in range(5):
    if i%2 == 0:
        number = 2*i + 1
        for j in range(5+i):
            if i+j >= 4:
                print(number, end='')
                number = number - 1
            else:
                print('-', end='')
        print()
    else:
        number = 1
        for j in range(5+i):
            if i+j >= 4:
                print(number, end='')
                number = number + 1
            else:
                print('-', end='')
        print()
```

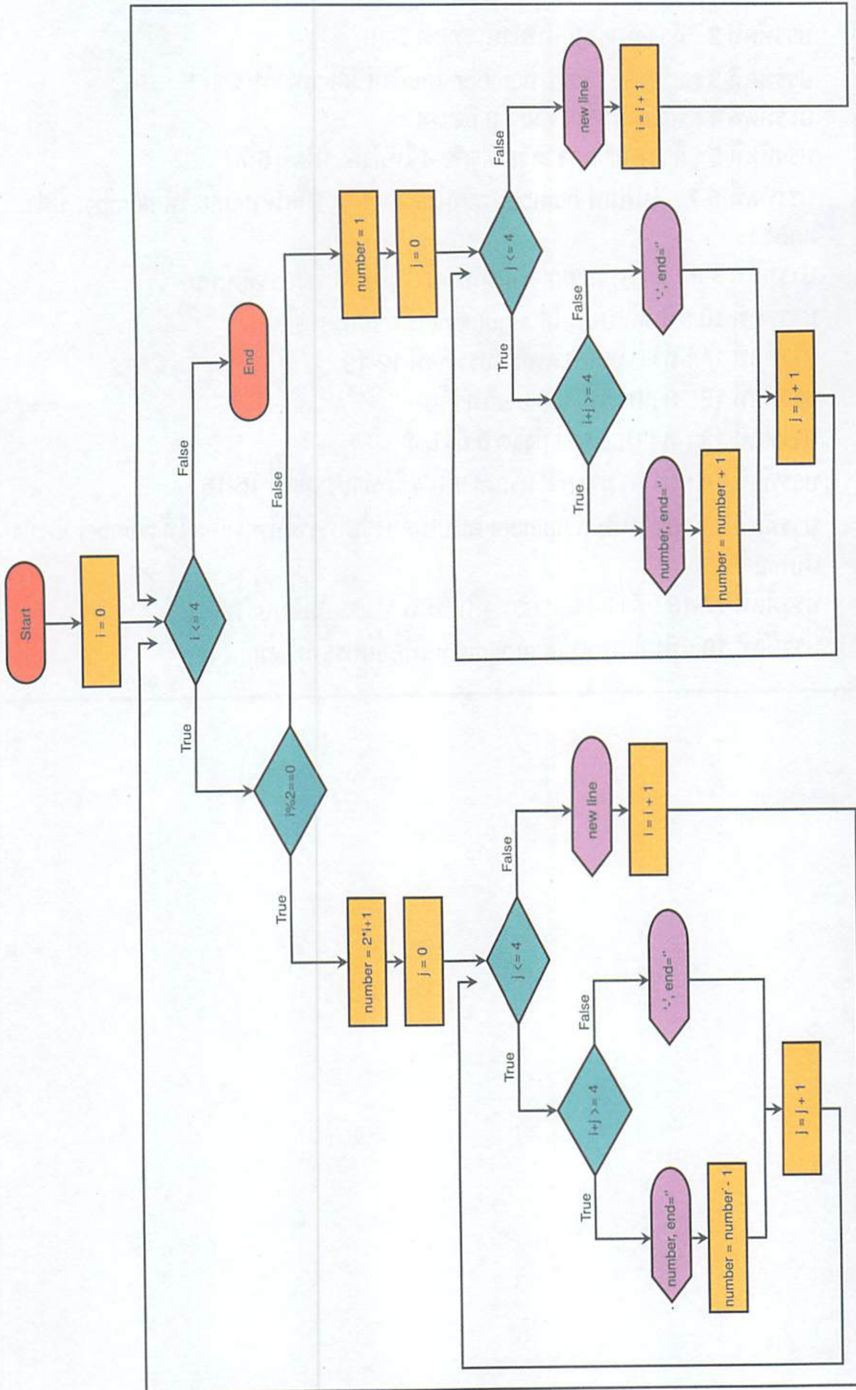
Output :

```
----1
---123
--54321
-1234567
987654321
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 0 ถึง 4
บรรทัดที่ 2 : ถ้า i เป็นเลขคู่ให้ทำบรรทัดที่ 3-10
บรรทัดที่ 3 : สร้างตัวแปรชื่อ number และกำหนดค่าเท่ากับ 2*i+1
บรรทัดที่ 4 : พิจารณาค่า j ตั้งแต่ 0 ถึง i+4
บรรทัดที่ 5 : ถ้า i+j มากกว่าหรือเท่ากับ 4 ให้ทำบรรทัดที่ 6-7
บรรทัดที่ 6-7 : ให้พิมพ์ number แบบไม่ต้องขึ้นบรรทัดใหม่ และ ให้ number มีค่าลดลง 1
บรรทัดที่ 8-9 : ถ้า i+j น้อยกว่า 4 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 10 : พิมพ์แบบไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่
บรรทัดที่ 11 : ถ้า i เป็นเลขคี่ให้ทำบรรทัดที่ 12-19
บรรทัดที่ 12 : กำหนดให้ number เท่ากับ 1
บรรทัดที่ 13 : พิจารณาค่า j ตั้งแต่ 0 ถึง i+4
บรรทัดที่ 14 : ถ้า i+j มากกว่าหรือเท่ากับ 4 ให้ทำบรรทัดที่ 15-16
บรรทัดที่ 15-16 : ให้พิมพ์ number แบบไม่ต้องขึ้นบรรทัดใหม่ และ ให้ number มีค่าเพิ่มขึ้น 1
บรรทัดที่ 17-18 : ถ้า i+j น้อยกว่า 4 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 19 : พิมพ์แบบไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

เฉลยข้อที่ 115.



12

Input :

```
for i in range(5):
    if i%2 == 0:
        number = 2*i + 1
        for j in range(5):
            if i+j >= 4:
                print(number, end='')
                number = number - 1
            else:
                print('-', end='')
        print()
    else:
        number = 1
        for j in range(5):
            if i+j >= 4:
                print(number, end='')
                number = number + 1
            else:
                print('-', end='')
        print()
```

Output :

```
----1
---12
--543
-1234
98765
```

เฉลยละเอียด

บรรทัดที่ 1 : พิจารณาค่า i ตั้งแต่ 0 ถึง 4
บรรทัดที่ 2 : ถ้า i เป็นเลขคู่ให้ทำบรรทัดที่ 3-10
บรรทัดที่ 3 : สร้างตัวแปรชื่อ number และกำหนดค่าเท่ากับ 2*i+1
บรรทัดที่ 4 : พิจารณาค่า j ตั้งแต่ 0 ถึง 4
บรรทัดที่ 5 : ถ้า i+j มากกว่าหรือเท่ากับ 4 ให้ทำบรรทัดที่ 6-7
บรรทัดที่ 6-7 : ให้พิมพ์ number แบบไม่ต้องขึ้นบรรทัดใหม่ และ ให้ number มีค่าลดลง 1
บรรทัดที่ 8-9 : ถ้า i+j น้อยกว่า 4 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 10 : พิมพ์แบบไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่
บรรทัดที่ 11 : ถ้า i เป็นเลขคี่ให้ทำบรรทัดที่ 12-19
บรรทัดที่ 12 : กำหนดให้ number เท่ากับ 1
บรรทัดที่ 13 : พิจารณาค่า j ตั้งแต่ 0 ถึง 4
บรรทัดที่ 14 : ถ้า i+j มากกว่าหรือเท่ากับ 4 ให้ทำบรรทัดที่ 15-16
บรรทัดที่ 15-16 : ให้พิมพ์ number แบบไม่ต้องขึ้นบรรทัดใหม่ และ ให้ number มีค่าเพิ่มขึ้น 1
บรรทัดที่ 17-18 : ถ้า i+j น้อยกว่า 4 ให้พิมพ์ '-' แบบไม่ขึ้นบรรทัดใหม่
บรรทัดที่ 19 : พิมพ์แบบไม่ใส่ argument เพื่อขึ้นบรรทัดใหม่

13 FUNCTION


สแกนเพื่อเรียนวิดีโอเนื้อหา
และเฉลยแบบฝึกหัดบทที่ 13


สแกนเพื่อรับ Code บทที่ 13

13.1 Problem

จงเขียน code ของโจทย์ต่อไปนี้

1. เขียนฟังก์ชันเพื่อบวกเลข 2 ตัว และ return ผลลัพธ์ออกมา
   กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ a และ b ที่เป็นจำนวนใด ๆ

2. เขียนฟังก์ชันเพื่อบวกเลข 3 ตัว และ return ผลลัพธ์ออกมา
   กำหนดให้ฟังก์ชันนี้รับ parameter 3 ตัว คือ a, b และ c ที่เป็นจำนวนใด ๆ

3. เขียนฟังก์ชันเพื่อยกกำลังสองเลข 1 ตัว และ return ผลลัพธ์ออกมา
   กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ a ที่เป็นจำนวนใด ๆ

4. เขียนฟังก์ชันเพื่อยกกำลังสองสมาชิกทุกตัวใน list และ return list ที่เป็นผลลัพธ์จากการยกกำลังสองออกมา
   กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ listA ที่มีสมาชิกเป็นจำนวนใด ๆ

   ตัวอย่างที่ 1 : สมมติให้ listA = [1, 2, 3, 4, 5] จะ return = [1, 4, 9, 16, 25]

5. เขียนฟังก์ชันเพื่อหาหาผลบวกกำลังสองของสมาชิกใน list และ return ผลลัพธ์ออกมา
   กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ listA ที่มีสมาชิกเป็นจำนวนใด ๆ

6. เขียนฟังก์ชันเพื่อหาผลบวกของจำนวนเต็มที่เป็นสมาชิกใน list และ return ผลลัพธ์ออกมา
   กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ listA ที่มีสมาชิกเป็นข้อมูลใด ๆ

7. เขียนฟังก์ชันเพื่อหาผลบวกของจำนวนจริงที่เป็นสมาชิกใน list และ return ผลลัพธ์ออกมา
   กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ listA ที่มีสมาชิกเป็นข้อมูลใด ๆ

8. เขียนฟังก์ชันเพื่อต่อสายอักขระ 2 สายเข้าด้วยกัน และ return ผลลัพธ์ออกมา
   กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ str1 และ str2 ที่เป็นสายอักขระใด ๆ

9. เขียนฟังก์ชันเพื่อต่อสายอักขระ 3 สายเข้าด้วยกัน และ return ผลลัพธ์ออกมา
   กำหนดให้ฟังก์ชันนี้รับ parameter 3 ตัว คือ str1, str2 และ str3 ที่เป็นสายอักขระใด ๆ

10. เขียนฟังก์ชันเพื่อลบคำจากสายอักขระ และ return ผลลัพธ์ออกมา
    กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ str1 และ str2 ที่เป็นสายอักขระใด ๆ

    ตัวอย่างที่ 1 : สมมติให้ str1 = 'onetwothree' และ str2 = 'two' จะ return 'onethree'

11. เขียนฟังก์ชันเพื่อสร้างคำซ้ำ และ return ผลลัพธ์ออกมา
    กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ str1 และ k ที่เป็นสายอักขระและจำนวนเต็มใด ๆ

    ตัวอย่างที่ 1 : สมมติให้ str1 = 'one' และ k = 3 จะ return 'one-one-one'

12. เขียนฟังก์ชันเพื่อหาความยาวรวมของสายอักขระที่เป็นสมาชิกใน list และ return ผลลัพธ์ออกมา
    กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ listA ที่มีสมาชิกเป็นจำนวนใด ๆ

13

13. เขียนฟังก์ชันเพื่อตัดอักขระพิเศษออกจากสายอักขระ และ return ผลลัพธ์ออกมา (กำหนดให้อักขระพิเศษ คือ !@#\$%,)
กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ str1 ที่เป็นสายอักขระใด ๆ

ตัวอย่างที่ 1 : สมมติให้ str1 = '@chicken' จะ return 'chicken'

ตัวอย่างที่ 2 : สมมติให้ str1 = '1,2%3$' จะ return '123'

14. เขียนฟังก์ชันเพื่อหาระยะห่างระหว่าง 2 จุดบนเส้นจำนวนจริง และ return ผลลัพธ์ออกมา **(hint: ระยะห่างระหว่าง 2 จุด = |x1 - x2|)**
กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ x1 และ x2 ที่เป็นจำนวนใด ๆ

15. เขียนฟังก์ชันเพื่อถ้าหากำไรขั้นต้น และ return ผลลัพธ์ออกมา
กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ sales (ยอดขาย) และ cogs (ต้นทุน) ที่เป็นจำนวนบวกใด ๆ

16. เขียนฟังก์ชันเพื่อลดราคาสินค้า และ return ราคาหลังได้รับส่วนลดแล้ว
กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ price ที่เป็นจำนวนบวกใด ๆ และ discount ที่เป็น % ซึ่งอยู่ในช่วง 0 - 100

17. เขียนฟังก์ชันกบกระโดด โดยรับ parameter 2 ตัว ได้แก่ d ซึ่งเป็นระยะทางที่กบต้องการกระโดด และ s ซึ่งเป็นระยะที่กระโดดได้ 1 ครั้งของกบ ให้ return จำนวนครั้งน้อยที่สุดที่กบต้องกระโดดเพื่อเดินทางให้ได้มากกว่าหรือเท่ากับ d

18. เขียนฟังก์ชันเพื่อค้นหาตำแหน่งใน list โดยรับ parameter 2 ตัว ได้แก่ listA ซึ่งเป็น list ที่เก็บข้อมูลใด ๆ (ไม่มีข้อมูลซ้ำ) และ k ซึ่งเป็นข้อมูลที่ต้องการค้นหา ให้ return ตำแหน่งของ listA ที่มี k ปรากฏ ถ้าไม่มี k ปรากฏให้ return -1

19. เขียนฟังก์ชันสำหรับทำงานในเครื่องแลกเหรียญอัตโนมัติ โดยรับ parameter 1 ตัว ได้แก่ amount ซึ่งเป็นจำนวนเต็มบวกใด ๆ ให้ return dictionary ที่มี key เป็น '10', '5', '2', '1' และ value เป็นจำนวนเหรียญชนิดนั้น ๆ โดยจะต้องแลกเหรียญที่มีค่ามากให้เสร็จก่อน

ตัวอย่างที่ 1 : สมมติให้ amount = 40 จะ return {'10' : 4, '5': 0, '2' : 0, '1', 0}
ตัวอย่างที่ 2 : สมมติให้ amount = 48 จะ return {'10' : 4, '5': 1, '2' : 1, '1', 1}

20. เขียนฟังก์ชันเพื่อคำนวณเงินทุนสะสม โดยรับ parameter 3 ตัว ได้แก่

- เงินลงทุนเริ่มต้น (PV)
- อัตราผลตอบแทนในรูป % ต่อปี (r)
- จำนวนปีที่ลงทุน (n)

ให้ return เงินลงทุนเมื่อผ่านไปแล้ว n ปี (คิดผลตอบแทนแบบทบต้น)

ตัวอย่างที่ 1 : สมมติให้ PV = 10000, r = 20, n = 1 จะ return 12000

ตัวอย่างที่ 2 : สมมติให้ PV = 5000, r = 15, n = 3 จะ return 7604.375

21. เขียนฟังก์ชันเพื่อหาอัตราผลตอบแทนในรูป % ต่อปี โดยรับ parameter 3 ตัว ได้แก่

- เงินลงทุนเริ่มต้น (PV)
- เงินลงทุนเมื่อผ่านไป n ปี (FV)
- จำนวนปีที่ลงทุน (n)

ให้ return ผลตอบแทนในรูป % ต่อปี

ตัวอย่างที่ 1 : สมมติให้ PV = 10000, FV = 12000, n = 1 จะ return 20

ตัวอย่างที่ 2 : สมมติให้ PV = 5000, FV = 7604.375, n = 3 จะ return 15

22. เขียนฟังก์ชัน Calculator(a, b, op) โดยรับ parameter 3 ตัว ได้แก่ a, b ซึ่งเป็นจำนวนใด ๆ และ op ซึ่งเป็นสายอักขระที่เป็นไปได้ 4 แบบ คือ '+', '-', '*', '/'

- ถ้า op เป็น '+' ให้ return a + b
- ถ้า op เป็น '-' ให้ return a - b
- ถ้า op เป็น '*' ให้ return a*b
- ถ้า op เป็น '/' ให้ return a/b

23. เขียนฟังก์ชันเพื่อหาความยาวด้านตรงข้ามมุมฉาก และ return ผลลัพธ์ออกมา กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ a และ b ที่เป็นด้านประกอบมุมฉาก

24. เขียนฟังก์ชันเพื่อหาระยะห่างระหว่างจุด 2 จุดในระนาบ 2 มิติ และ return ผลลัพธ์ออกมา (hint: ระยะห่างระหว่าง 2 จุด = sqrt((x1 - x2)**2 + (y1 - y2)**2)
กำหนดให้ฟังก์ชันนี้รับ parameter 4 ตัว คือ x1, y1, x2, y2 ที่เป็นจำนวนใด ๆ

25. เขียนฟังก์ชันเพื่อหาระยะห่างระหว่างจุด 2 จุดในระนาบ n มิติ และ return ผลลัพธ์ออกมา (hint: euclidean distance)
กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ P1 และ P2 ที่เป็น tuple ความยาว n

26. เขียนฟังก์ชันสำหรับ shift สมาชิกใน list ไปทางขวา k step โดยรับ parameter 2 ตัว ได้แก่ listA ซึ่งเป็น list ที่เก็บข้อมูลใด ๆ และ k ซึ่งเป็นจำนวนครั้งที่ต้องการให้ shift ไปทางขวา ให้ return listA ที่ผ่านการ shift ออกมา

> **ตัวอย่างที่ 1** : สมมติให้ listA = [1, 2, 3, 4, 5] และ k = 1 จะ return [5, 1, 2, 3, 4]
>
> **ตัวอย่างที่ 2** : สมมติให้ listA = [1, 2, 3, 4, 5] และ k = 2 จะ return [4, 5, 1, 2, 3]

27. เขียนฟังก์ชันสำหรับ shift สมาชิกใน list ไปทางซ้าย k step โดยรับ parameter 2 ตัว ได้แก่ listA ซึ่งเป็น list ที่เก็บข้อมูลใด ๆ และ k ซึ่งเป็นจำนวนครั้งที่ต้องการให้ shift ไปทางซ้าย ให้ return listA ที่ผ่านการ shift ออกมา

> **ตัวอย่างที่ 1** : สมมติให้ listA = [1, 2, 3, 4, 5] และ k = 1 จะ return [2, 3, 4, 5, 1]
>
> **ตัวอย่างที่ 2** : สมมติให้ listA = [1, 2, 3, 4, 5] และ k = 2 จะ return [3, 4, 5, 1, 2]

13

28. เขียนฟังก์ชันเพื่อแปลงเลขฐานใด ๆ เป็นเลขฐาน 10 โดยรับ parameter 2 ตัว คือ s ซึ่งเป็นสายอักขระที่เก็บเลขฐานใด ๆ และ b ซึ่งเป็นจำนวนเต็มบวกที่บ่งบอกเลขฐานของ s ให้ return จำนวนที่เป็นเลขฐาน 10 ของ s

> **ตัวอย่างที่ 1** : สมมติให้ s = '10110' และ b = 2 จะ return 22
>
> **ตัวอย่างที่ 2** : สมมติให้ s = '71A' และ b = 16 จะ return 1818

29. เขียนฟังก์ชันเพื่อแปลงเลขฐาน 10 เป็นเลขฐาน 2 โดยรับ parameter 1 ตัว คือ x ซึ่งเป็นจำนวนเต็มบวกใด ๆ ให้ return เลขฐาน 2 ของ x ในรูปแบบของสายอักขระ

30. เขียนฟังก์ชันเพื่อแปลงเลขฐาน 10 เป็นเลขฐาน 16 โดยรับ parameter 1 ตัว คือ x ซึ่งเป็นจำนวนเต็มบวกใด ๆ ให้ return เลขฐาน 16 ของ x ในรูปแบบของสายอักขระ

31. เขียนฟังก์ชันเพื่อหาค่าต่ำสุด และค่าสูงสุด ให้ return ผลลัพธ์ทั้ง 2 ออกมา
กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ listA ที่มีสมาชิกเป็นจำนวนใด ๆ

32. เขียนฟังก์ชันเพื่อหาค่าเฉลี่ย, มัธยฐาน และฐานนิยม จากนั้นให้ return ผลลัพธ์ทั้ง 3 ออกมา
กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ listA ที่มีสมาชิกเป็นจำนวนใด ๆ

13

33. เขียนฟังก์ชันเพื่อตรวจสอบจำนวนเฉพาะ

    - ถ้าเป็นจำนวนเฉพาะ ให้ return True
    - ถ้าไม่เป็นจำนวนเฉพาะ ให้ return False

    กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ num ที่เป็นจำนวนเต็มบวกใด ๆ

34. เขียนฟังก์ชันเพื่อตรวจสอบความเป็นจำนวนเฉพาะของสมาชิกทุกตัวใน list และ return list ที่เป็นผลลัพธ์จากการตรวจสอบออกมา
กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ listA ที่มีสมาชิกเป็นจำนวนเต็มใด ๆ

> **ตัวอย่างที่ 1** : สมมติให้ listA = [1, 2, 3, 4, 5] จะ return = [False, True, True, False, True]

35. เขียนฟังก์ชันสำหรับวาดรูปต่อไปนี้โดยรับ parameter 1 ตัว คือ n

ตัวอย่างที่ 1 : n = 5

```
x---x
-x-x-
--x--
-x-x-
x---x
```

ตัวอย่างที่ 2 : n = 6

```
x----x
-x--x-
--xx--
--xx--
-x--x-
x----x
```

36. เขียนฟังก์ชันสำหรับวาดรูปต่อไปนี้โดยรับ parameter 1 ตัว คือ n

ตัวอย่างที่ 1 : n = 5

```
--x--
-x-x-
x---x
-x-x-
--x--
```

ตัวอย่างที่ 2 : n = 6

```
--xx--
-x--x-
x----x
-x--x-
--xx--
```

37. เขียนฟังก์ชันสำหรับวาดรูปต่อไปนี้ โดยไม่ต้องรับ parameter

```
---x
--xxx
-xxxxx
xxxxxxx
```

38. เขียนฟังก์ชันเพื่อหาผลบวก และผลคูณใน index ที่ตรงกันของ 2 list และ return ผลลัพธ์ทั้ง 2 ออกมา

กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ listA และ listB ที่มีสมาชิกเป็นจำนวนเต็มใด ๆ

ตัวอย่างที่ 1 : สมมติให้ listA = [1, 2, 3] และ listB = [6, 7, 8] จะ return [7, 8, 9] และ [6, 14, 24]

13

39. เขียนฟังก์ชันเพื่อหา list ของ key และ value ของ dictionary และ return ผลลัพธ์ทั้ง 2 ออกมา
กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ dictA ที่มีสมาชิกเป็นข้อมูลใด ๆ

> **ตัวอย่างที่ 1** : สมมติให้ dictA = {'name' : 'John', 'age' : 32} จะ return ['name', 'age'] และ ['John', 32]

40. เขียนฟังก์ชันเพื่อหาผลบวกสะสมไล่ตาม index และ return ผลลัพธ์ออกมา
กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ listA ที่มีสมาชิกเป็นจำนวนใด ๆ

> **ตัวอย่างที่ 1** : สมมติให้ listA = [1, 2, 3, 4, 5] จะ return [1, 3, 6, 10, 15]

41. เขียนฟังก์ชันเพื่อตรวจสอบด้านประกอบของสามเหลี่ยม โดยรับ parameter 3 ตัว ได้แก่ a, b, c ซึ่งเป็นจำนวนบวกใด ๆ ที่เป็นความยาวด้านทั้ง 3 ของสามเหลี่ยม ให้ตรวจสอบว่า a, b, c สามารถเป็นความยาวด้านของสามเหลี่ยมได้หรือไม่

- ถ้าได้ ให้ return True
- ถ้าไม่ได้ ให้ return False

42. เขียนฟังก์ชันเพื่อแปลงสายอักขระจาก lower case เป็น upper case และจาก upper case เป็น lower case และ return ผลลัพธ์ออกมา
กำหนดให้ฟังก์ชันนี้รับ parameter 1 ตัว คือ word ที่เป็นสายอักขระใด ๆ

13

43. เขียนฟังก์ชันเพื่อลบสายอักขระที่มีความยาวเกินกว่าค่าที่กำหนดออกจาก list และ return ผลลัพธ์ออกมา
กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ listA ที่มีสมาชิกเป็นสายอักขระใด ๆ และ n ที่เป็นจำนวนเต็มใด ๆ

> **ตัวอย่างที่ 1** : สมมติให้ listA= ['name', 'n', 'me'] และ n = 3 จะ return ['n', 'me']

44. เขียนฟังก์ชันสำหรับหา derivatives ของพหุนาม โดยรับ parameter 1 ตัว คือ listA ซึ่งเป็น list ที่เก็บสัมประสิทธิ์หน้าพจของพหุนาม โดยที่ index ของ listA จะบ่งบอกพจน์ของเลขชี้กำลัง ให้ return list ที่มีความยาวเท่ากับ listA ที่เก็บสัมประสิทธิ์จาก derivatives ของพหุนามที่รับมา

ตัวอย่างที่ 1 : listA = [1, 2, 2] (แทนพหุนามของ $2x^2+2x+1$) จะ return [2, 4, 0] (แทนพหุนามของ 4x + 2)

ตัวอย่างที่ 2 : listA = [2, 3, 0, 1] (แทนพหุนามของ x^3+3x+2) จะ return [3, 0, 3, 0] (แทนพหุนามของ $3x^2+3$)

45. เขียนฟังก์ชัน reshape matrix จาก 1D ไปเป็น 2D

กำหนดให้ฟังก์ชันนี้รับ parameter 2 ตัว คือ listA และ tuple 2 มิติ เพื่อบอก shape ใหม่

13

13.2 Solution

เฉลยข้อที่ 1.

Input :

```
1 def addition(a, b):
2     return a + b
3
4 addition(18, 32)
```

Output :

```
50
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ addition ที่รับ parameter เป็น a และ b
บรรทัดที่ 2 : return ค่าผลบวกของ a และ b

เฉลยข้อที่ 2.

Input :

```
1 def addition3(a, b, c):
2     return a + b + c
3
4 addition3(10, 20, 25)
```

Output :

```
55
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ addition3 ที่รับ parameter เป็น a, b และ c
บรรทัดที่ 2 : return ค่าผลบวกของ a, b และ c

เฉลยข้อที่ 3.

Input :

```
1 def square(a):
2     return a**2
3
4 square(25)
```

Output :

```
625
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ square ที่รับ parameter เป็น a
บรรทัดที่ 2 : return ค่ากำลังสองของ a

เฉลยข้อที่ 4.

Input :

```
1 def square_list(listA):
2     square = []
3     for a in listA:
4         square.append(a**2)
5     return square
6
7 square_list([1, 2, 3, 4, 5])
```

Output :

```
[1, 4, 9, 16, 25]
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ square_list ที่รับ parameter เป็น listA
บรรทัดที่ 2 : สร้างตัวแปรชื่อ square และกำหนดให้เป็น empty list
บรรทัดที่ 3-4 : นำสมาชิกแต่ละตัวใน listA มายกกำลังสอง แล้วนำไป append ใน square
บรรทัดที่ 5 : return square ออกมา

เฉลยข้อที่ 5.

Input :

```
1 def sum_square_list(listA):
2     sumx = 0
3     for a in listA:
4         sumx = sumx + a**2
5     return sumx
6
7 sum_square_list([1, 2, 3, 4, 5])
```

Output :

```
55
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ sum_square_list ที่รับ parameter เป็น listA
บรรทัดที่ 2 : สร้างตัวแปรชื่อ sumx และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3-4 : นำสมาชิกแต่ละตัวใน listA มายกกำลังสอง แล้วนำไปบวกเพิ่มที่ sumx
บรรทัดที่ 5 : return sumx ออกมา

เฉลยข้อที่ 6.

Input :

```
def sum_int(listA):
    sumx = 0
    for a in listA:
        if type(a) == int:
            sumx = sumx + a
    return sumx

sum_int([1.1, 2, 3.3, 4, 5.5, 6])
```

Output :

```
12
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ sum_int ที่รับ parameter เป็น listA
บรรทัดที่ 2 : สร้างตัวแปรชื่อ sumx และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3 : พิจารณาสมาชิกแต่ละตัวของ listA ในชื่อตัวแปร a
บรรทัดที่ 4-5 : ถ้า type ของ a เป็น int ให้นำ a ไปบวกเพิ่มที่ sumx
บรรทัดที่ 6 : return sumx ออกมา

เฉลยข้อที่ 7.

Input :

```
1 def sum_float(listA):
2     sumx = 0
3     for a in listA:
4         if type(a) == float:
5             sumx = sumx + a
6     return sumx
7 
8 sum_float([1.1, 2, 3.3, 4, 5.5, 6])
```

Output :

```
9.9
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ sum_float ที่รับ parameter เป็น listA
บรรทัดที่ 2 : สร้างตัวแปรชื่อ sumx และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3 : พิจารณาสมาชิกแต่ละตัวของ listA ในตัวแปรชื่อ a
บรรทัดที่ 4-5 : ถ้า type ของ a เป็น float ให้นำ a ไปบวกเพิ่มที่ sumx
บรรทัดที่ 6 : return sumx ออกมา

เฉลยข้อที่ 8.

Input :

```
def concat_2string(str1, str2):
    return str1 + str2

concat_2string('Goo', 'gle')
```

Output :

```
'Google'
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ concat_2string ที่รับ parameter เป็น str1 และ str2
บรรทัดที่ 2 : return ผลบวกของ str1 และ str2 (การบวกกันของ string คือการนำมาต่อกัน)

เฉลยข้อที่ 9.

Input :

```
1 def concat_3string(str1, str2, str3):
2     return str1 + str2 + str3
3
4 concat_3string('A', 'ma', 'zon')
```

Output :

```
'Amazon'
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ concat_3string ที่รับ parameter เป็น str1, str2 และ str3
บรรทัดที่ 2 : return ผลบวกของ str1, str2 และ str3 (การบวกกันของ string คือการนำมาต่อกัน)

เฉลยข้อที่ 10.

13

Input :

```
1 def del_word(str1, str2):
2     return str1.replace(str2, '')
3
4 del_word('onetwothree', 'two')
```

Output :

```
'onethree'
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ del_word ที่รับ parameter เป็น str1 และ str2
บรรทัดที่ 2 : return str1 ที่ถูกแทนที่ str2 ด้วย empty string

เฉลยข้อที่ 11.

Input :

```
1 def repeat_word(str1, k):
2     word = (str1 + '-')
3     word_k = word*k
4     return word_k[:-1]
5
6 repeat_word('one', 3)
```

Output :

```
'one-one-one'
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ repeat_word ที่รับ parameter เป็น str1 และ k
บรรทัดที่ 2 : สร้างตัวแปรชื่อ word เพื่อเก็บสายอักขระของการนำ str1 มาต่อด้วย '-'
บรรทัดที่ 3 : สร้างตัวแปรชื่อ word_k เพื่อเก็บสายอักขระจากการนำ word มาต่อกัน k ครั้ง
บรรทัดที่ 4 : return word_k ที่ทำการตัดตำแหน่งสุดท้ายที่เป็น '-' ออกไป

เฉลยข้อที่ 12.

Input :

```
def sum_len(listA):
    sumx = 0
    for a in listA:
        if type(a) == str:
            sumx = sumx + len(a)
    return sumx

sum_len(['food', 'drink', 'snack', 1, 2, 3])
```

Output :

```
14
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ sum_len ที่รับ parameter เป็น listA
บรรทัดที่ 2 : สร้างตัวแปรชื่อ sumx และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3 : พิจารณาสมาชิกแต่ละตัวของ listA ในชื่อตัวแปร a
บรรทัดที่ 4-5 : ถ้า type ของ a เป็น str ให้นำความยาวของ a ไปบวกเพิ่มที่ sumx
บรรทัดที่ 6 : return sumx ออกมา

เฉลยข้อที่ 13.

Input :

```
1 def del_special_char(str1):
2     special_char = '!@#\$%,'
3     for c in special_char:
4         str1 = str1.replace(c, '')
5     return str1
6 
7 del_special_char('chicken%')
```

Output :

```
'chicken'
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ del_special_char ที่รับ parameter เป็น str1 **บรรทัดที่ 2** : สร้างตัวแปรชื่อ special_char เพื่อเก็บอักขระพิเศษ
บรรทัดที่ 3 : พิจารณาอักขระพิเศษแต่ละตัว ในชื่อตัวแปร c
บรรทัดที่ 4 : แทนที่อักขระพิเศษที่ปรากฏใน str1 ด้วย empty string
บรรทัดที่ 5 : return str1 ที่ลบอักขระพิเศษออกแล้ว

13

เฉลยข้อที่ 14.

Input :

```
1 def distance_R1(x1, x2):
2     return abs(x1 - x2)
3 
4 distance_R1(32, -18)
```

Output :

```
50
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ distance_R1 ที่รับ parameter เป็น x1 และ x2
บรรทัดที่ 2 : return ค่า absolute ของผลต่างระหว่าง x1 และ x2

เฉลยข้อที่ 15.

Input :

```
1 def calculate_profit(sales, cogs):
2     return sales - cogs
3
4 print('Gross Profit =', calculate_profit(5000, 4000)
    ↪ )
```

Output :

```
Gross Profit = 1000
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ calculate_profit ที่รับ parameter เป็น sales และ cogs
บรรทัดที่ 2 : return กำไรขั้นต้นจากการหา sales - cogs

เฉลยข้อที่ 16.

Input :

```
1 def sale(price, discount):
2     return price*(100 - discount)/100
3
4 print('Discounted Price =', sale(1000, 20))
```

Output :

```
Discounted Price = 800.0
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ sale ที่รับ parameter เป็น price และ discount
บรรทัดที่ 2 : return ราคาหลังหักส่วนลดแล้ว

เฉลยข้อที่ 17.

Input :

```
1 def frog_jump(d, s):
2     import math
3     return math.ceil(d/s)
4 
5 frog_jump(10, 2)
```

Output :

```
5
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ frog_jump ที่รับ parameter เป็น d และ s
บรรทัดที่ 2 : import library ชื่อ math
บรรทัดที่ 3 : return ผลหารระหว่างระยะทางที่ต้องการกระโดด กับ ระยะทางที่ได้จากการกระโดด 1 ครั้ง แล้วทำการปัดเศษขึ้น

เฉลยข้อที่ 18.

Input :

```
def find_index(listA, k):
    if k in listA:
        index_list = []
        n = len(listA)
        for i in range(n):
            if listA[i] == k:
                index_list.append(i)
        return index_list
    else:
        return - 1

find_index([2, 4, 6, 8, 12], 4)
```

Output :

```
[1]
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ find_index ที่รับ parameter เป็น listA และ k
บรรทัดที่ 2 : ถ้า k อยู่ใน listA ให้ทำบรรทัดที่ 3-8
บรรทัดที่ 3 : สร้างตัวแปรชื่อ index_list และกำหนดให้เป็น empty list
บรรทัดที่ 4 : สร้างตัวแปรชื่อ n และกำหนดค่าเท่ากับความยาวของ listA
บรรทัดที่ 5 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 6-7 : ถ้าสมาชิกตำแหน่งที่ i ของ listA มีค่าเท่ากับ k ให้นำ i ไป append ใน index_list
บรรทัดที่ 8 : return index_list ออกมา
บรรทัดที่ 9-10 : ถ้า k ไม่อยู่ใน listA ให้ return -1

เฉลยข้อที่ 19.

Input :

```
def coin_changer(amount):
    coin_list = ['10', '5', '2', '1']
    coin_dict = {'10' : 0, '5' : 0, '2' : 0, '1' : 0}
    for c in coin_list:
        if amount == 0:
            break
        else:
            n_coin = amount//int(c)
            coin_dict[c] = n_coin
            amount = amount - n_coin*int(c)
    return coin_dict

coin_changer(58)
```

Output :

```
{'10': 5, '5': 1, '2': 1, '1': 1}
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ coin_changer ที่รับ parameter เป็น amount
บรรทัดที่ 2 : สร้างตัวแปรชื่อ coin_list เพื่อเก็บมูลค่าของเหรียญแต่ละประเภท
บรรทัดที่ 3 : สร้างตัวแปรชื่อ coin_dict เพื่อเก็บ dictionary ที่มี key เป็นมูลค่าของเหรียญ และ value เป็นจำนวนของเหรียญแต่ละประเภท
บรรทัดที่ 4 : พิจารณาสมาชิกแต่ละตัวใน coin_list ในชื่อตัวแปร c
บรรทัดที่ 5-6 : ถ้า amount เท่ากับ 0 ให้ break ออกจาก loop
บรรทัดที่ 7 : ถ้า amount ไม่เป็น 0 ให้ทำบรรทัดที่ 8-10
บรรทัดที่ 8 : สร้างตัวแปรชื่อ n_coin เพื่อคำนวณจำนวนเหรียญที่ต้องใช้สำหรับแลกเหรียญ c
บรรทัดที่ 9 : นำจำนวนเหรียญไปเก็บใน coin_dict
บรรทัดที่ 10 : นำมูลค่าที่แลกเหรียญ c ไปหักลบที่ amount
บรรทัดที่ 11 : return coin_dict ออกมา

เฉลยข้อที่ 20.

Input :

```
1 def calculate_FV(PV, r, n):
2     FV = PV*(1 + r/100)**n
3     return FV
4 
5 calculate_FV(1000, 20, 2)
```

Output :

```
25
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ calculate_FV ที่รับ parameter เป็น PV, r และ n
บรรทัดที่ 2 : สร้างตัวแปรชื่อ FV สำหรับหาเงินลงทุนรวม ซึ่งคำนวณได้ตามสูตรข้างต้น
บรรทัดที่ 3 : return FV ออกมา

เฉลยข้อที่ 21.

Input :

```
1 def calculate_r(PV, FV, n):
2     r = ((FV/PV)**(1/n) - 1)*100
3     return r
4 
5 calculate_r(1000, 1440, 2)
```

Output :

```
19.999999999999996
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ calculate_r ที่รับ parameter เป็น PV, FV และ n

บรรทัดที่ 2 : สร้างตัวแปรชื่อ r สำหรับหาอัตราผลตอบแทน ซึ่งคำนวณได้ตามสูตรข้างต้น

บรรทัดที่ 3 : return r ออกมา

เฉลยข้อที่ 22.

Input :

```
def Calculator(a, b, op):
    if op == '+':
        return a + b
    elif op == '-':
        return a - b
    elif op == '*':
        return a*b
    elif op == '/':
        return a/b

Calculator(5, 2, '*')
```

Output :

```
10
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ Calculator ที่รับ parameter เป็น a, b และ op
บรรทัดที่ 2-3 : ถ้า op เป็นเครื่องหมายบวก ให้ return a+b
บรรทัดที่ 4-5 : ถ้า op เป็นเครื่องหมายลบ ให้ return a-b
บรรทัดที่ 6-7 : ถ้า op เป็นเครื่องหมายคูณ ให้ return a*b
บรรทัดที่ 8-9 : ถ้า op เป็นเครื่องหมายหาร ให้ return a/b

เฉลยข้อที่ 23.

Input :

```
1 def calculate_c(a, b):
2     c = (a**2 + b**2)**(1/2)
3     return c
4 
5 calculate_c(3, 4)
```

Output :

```
5.0
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ calculate_c ที่รับ parameter เป็น a และ b
บรรทัดที่ 2 : สร้างตัวแปรชื่อ c สำหรับหาความยาวด้านตรงข้ามมุมฉาก ซึ่งคำนวณได้ตามสูตรข้างต้น
บรรทัดที่ 3 : return c ออกมา

เฉลยข้อที่ 24.

Input :

```
def distance_R2(x1, y1, x2, y2):
    distance = ((x1 - x2)**2 + (y1 - y2)**2)**(1/2)
    return distance

distance_R2(3, 4, 6, 8)
```

Output :

```
5.0
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ distance_R2 ที่รับ parameter เป็น x1, y1, x2 และ y2
บรรทัดที่ 2 : สร้างตัวแปรชื่อ distance สำหรับหาระยะห่าง ซึ่งคำนวณได้ตามสูตรข้างต้น
บรรทัดที่ 3 : return distance ออกมา

เฉลยข้อที่ 25.

Input :

```
1 def distance_Rn(P1, P2):
2     sum_square = 0
3     n = len(P1)
4     for i in range(n):
5         sum_square = sum_square + (P1[i] - P2[i])**2
6     distance = sum_square**(1/2)
7     return distance
8 
9 distance_Rn((1, 2, 2, 1), (3, 4, 0, -1))
```

Output :

```
4.0
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ distance_Rn ที่รับ parameter เป็น P1 และ P2
บรรทัดที่ 2 : สร้างตัวแปรชื่อ sum_square และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3 : สร้างตัวแปรชื่อ n และกำหนดค่าเท่ากับความยาวของ P1
บรรทัดที่ 4 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 5 : นำผลต่างของ P1 และ P2 ในตำแหน่งที่ i ไปยกกำลังสองแล้วนำไปบวกเพิ่มที่ sum_square
บรรทัดที่ 6 : สร้างตัวแปรชื่อ distance สำหรับหาระยะห่าง ซึ่งคำนวณได้ตามสูตรข้างต้น
บรรทัดที่ 7 : return distance ออกมา

เฉลยข้อที่ 26.

Input :

```
1 def shift_right(listA, k):
2     k = k%len(listA)
3     return listA[-k:] + listA[:-k]
4
5 shift_right([1, 2, 3, 4, 5], 1)
```

Output :

```
[5, 1, 2, 3, 4]
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ shift_right ที่รับ parameter เป็น listA และ k
บรรทัดที่ 2 : ทำการ mod k ด้วยความยาวของ listA (เนื่องจากถ้าจำนวนครั้งที่ทำการ rotate ไปเท่ากับจำนวนสมาชิกใน listA จะทำให้สมาชิกกลับมาอยู่ที่เดิม)
บรรทัดที่ 3 : return listA ที่ทำการ rotate แล้ว

เฉลยข้อที่ 27.

Input :

```
1 def shift_left(listA, k):
2     k = k%len(listA)
3     return listA[k:] + listA[:k]
4
5 shift_left([1, 2, 3, 4, 5], 1)
```

Output :

```
[2, 3, 4, 5, 1]
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ shift_left ที่รับ parameter เป็น listA และ k
บรรทัดที่ 2 : ทำการ mod k ด้วยความยาวของ listA (เนื่องจากถ้าจำนวนครั้งที่ทำการ rotate ไปเท่ากับจำนวนสมาชิกใน listA จะทำให้สมาชิกกลับมาอยู่ที่เดิม)
บรรทัดที่ 3 : return listA ที่ทำการ rotate แล้ว

เฉลยข้อที่ 28.

Input :

```
1  def to_decimal(s, b):
2      converter = {'0' : 0, '1' : 1, '2' : 2, '3' : 3,
       ↪  '4' : 4, '5' : 5, '6' : 6, '7' : 7, '8' : 8,
       ↪  '9' : 9, 'A' : 10, 'B' : 11, 'C' : 12, 'D' :
       ↪  13, 'E' : 14, 'F' : 15}
3      s = s[::-1].upper()
4      decimal = 0
5      n = len(s)
6      for i in range(n):
7          decimal = decimal + converter[s[i]]*b**(i)
8      return decimal
9
10 to_decimal('71a', 16)
```

Output :

```
1818
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ to_decimal ที่รับ parameter เป็น s และ b
บรรทัดที่ 2 : สร้างตัวแปรชื่อ converter เพื่อแปลงค่า string ตั้งแต่ 0-F มาเป็นตัวเลข 0-15
บรรทัดที่ 3 : ทำการเรียง string กลับหลังเพื่อให้ทำการเริ่มคำนวณจากหลักหน่วยก่อน
บรรทัดที่ 4 : สร้างตัวแปรชื่อ decimal และกำหนดค่าเท่ากับ 0 เพื่อเก็บค่าที่แปลงเป็นฐานสิบแล้ว
บรรทัดที่ 5 : สร้างตัวแปรชื่อ n และกำหนดค่าเท่ากับความยาวของ s
บรรทัดที่ 6-7 : แปลงค่าในแต่ละหลักมาเป็นฐานสิบแล้วนำไปบวกรวมกัน
บรรทัดที่ 8 : return ค่าฐานสิบ

เฉลยข้อที่ 29.

Input :

```
1 def to_binary(x):
2     import math
3     n_bit = math.floor(math.log(x, 2))
4     binary = ''
5     for bit in range(n_bit, -1, -1):
6         s = x//int(2**bit)
7         binary = binary + str(s)
8         x = x - s*2**bit
9     return binary
10
11 to_binary(7)
```

Output :

```
'111'
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ to_binary ที่รับ parameter เป็น x
บรรทัดที่ 2 : import library ชื่อ math
บรรทัดที่ 3 : หาจำนวนบิตในการเก็บค่า binary และกำหนดค่าใส่ตัวแปรชื่อ n_bit
บรรทัดที่ 4 : สร้างตัวแปรชื่อ binary และกำหนดให้เป็น empty string
บรรทัดที่ 5 : พิจารณา bit ตั้งแต่ n_bit จนถึง 0 แบบถอยหลัง
บรรทัดที่ 6 : หาค่าของหลักที่ bit ด้วยการหารแบบปัดเศษลง
บรรทัดที่ 7 : นำค่าของหลักที่ bit ไปต่อกับ binary
บรรทัดที่ 8 : หักลบค่า x ออกตามค่าที่ได้คำนวณไปแล้ว
บรรทัดที่ 9 : return binary ออกมา

เฉลยข้อที่ 30.

Input :

```
1 def to_hexadecimal(x):
2     converter = {0 : '0', 1 : '1', 2 : '2', 3 : '3',
    ↪  4 : '4', 5 : '5', 6 : '6', 7 : '7', 8 : '8',
    ↪  9 : '9', 10 : 'A', 11 : 'B', 12 : 'C', 13 :
    ↪ 'D', 14 : 'E', 15 : 'F'}
3     import math
4     n_bit = math.floor(math.log(x, 16))
5     hexa = ''
6     for bit in range(n_bit, -1, -1):
7         s = x//int(16**bit)
8         hexa = hexa + converter[s]
9         x = x - s*16**bit
10    return hexa
11
12 to_hexadecimal(318)
```

Output :

```
'13E'
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ to_hexadecimal ที่รับ parameter เป็น x
บรรทัดที่ 2 : สร้างตัวแปรชื่อ converter เพื่อแปลงค่า string ตั้งแต่ 0-F มาเป็นตัวเลข 0-15
บรรทัดที่ 3 import library ชื่อ math
บรรทัดที่ 4 : หาจำนวนหลักในการเก็บค่า hexadecimal และกำหนดค่าใส่ตัวแปรชื่อ n_bit
บรรทัดที่ 5 : สร้างตัวแปรชื่อ hexa และกำหนดให้เป็น empty string
บรรทัดที่ 6 : พิจารณา bit ตั้งแต่ n_bit จนถึง 0 แบบถอยหลัง
บรรทัดที่ 7 : หาค่าของหลักที่ bit ด้วยการหารแบบปัดเศษลง
บรรทัดที่ 8 : นำค่าของหลักที่ bit แปลงด้วย converter แล้วนำไปต่อกับ hexa
บรรทัดที่ 9 : หักลบค่า x ออกตามค่าที่ได้คำนวณไปแล้ว
บรรทัดที่ 10 : return hexa ออกมา

เฉลยข้อที่ 31.

Input :

```
1  def find_min_max(listA):
2      minA = listA[0]
3      maxA = listA[0]
4      n = len(listA)
5      for i in range(1, n):
6          if listA[i] < minA:
7              minA = listA[i]
8          if listA[i] > maxA:
9              maxA = listA[i]
10     return minA, maxA
11
12 _min, _max = find_min_max([19, 1, 32, -18])
13 print('min =', _min)
14 print('max =', _max)
```

Output :

```
min = -18
max = 32
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ find_min_max ที่รับ parameter เป็น listA
บรรทัดที่ 2 : สร้างตัวแปรชื่อ minA และกำหนดค่าเท่ากับสมาชิกตัวแรกของ listA
บรรทัดที่ 3 : สร้างตัวแปรชื่อ maxA และกำหนดค่าเท่ากับสมาชิกตัวแรกของ listA
บรรทัดที่ 4 : สร้างตัวแปรชื่อ n และกำหนดค่าเท่ากับความยาวของ listA
บรรทัดที่ 5 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 6-7 : ถ้าสมาชิกตำแหน่งที่ i ของ listA มีค่าน้อยกว่า minA ให้ minA เท่ากับสมาชิกตัวนั้น
บรรทัดที่ 8-9 : ถ้าสมาชิกตำแหน่งที่ i ของ listA มีค่ามากกว่า maxA ให้ maxA เท่ากับสมาชิกตัวนั้น
บรรทัดที่ 10 : return minA และ maxA ออกมา

เฉลยข้อที่ 32.

Input :

```
1 def find_mean_median_mode(listA):
2     import statistics
3     meanA = statistics.mean(listA)
4     medianA = statistics.median(listA)
5     modeA = statistics.mode(listA)
6     return meanA, medianA, modeA
7
8 mean, median, mode = find_mean_median_mode([1, 2, 2,
     ↪ 3, 4])
9 print('mean =', mean)
10 print('median =', median)
11 print('mode =', mode)
```

Output :

```
mean = 2.4
median = 2
mode = 2
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ find_mean_median_mode ที่รับ parameter เป็น listA
บรรทัดที่ 2 : import library ชื่อ statistics
บรรทัดที่ 3 : สร้างตัวแปรชื่อ meanA สำหรับเก็บค่าเฉลี่ยของข้อมูล
บรรทัดที่ 4 : สร้างตัวแปรชื่อ medianA สำหรับเก็บค่ามัธยฐานของข้อมูล
บรรทัดที่ 5 : สร้างตัวแปรชื่อ modeA สำหรับเก็บค่าฐานนิยมของข้อมูล
บรรทัดที่ 6 : return meanA, medianA และ modeA ออกมา

เฉลยข้อที่ 33.

Input :

```
1  def isPrime(num):
2      count = 0
3      for i in range(1, num + 1):
4          if num%i == 0:
5              count = count + 1
6      if count == 2:
7          return True
8      else:
9          return False
10
11 isPrime(37)
```

Output :

```
True
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ isPrime ที่รับ parameter เป็น num
บรรทัดที่ 2 : สร้างตัวแปรชื่อ count และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3 : พิจารณาค่า i ตั้งแต่ 1 ถึง num
บรรทัดที่ 4-5 : ถ้า num หารด้วย i ลงตัว ให้เพิ่ม count ขึ้น 1
บรรทัดที่ 6-7 : ถ้า count เท่ากับ 2 ให้ return True
บรรทัดที่ 8-9 : ถ้า count ไม่เท่ากับ 2 ให้ return False

เฉลยข้อที่ 34.

Input :

```
def isPrime_list(listA):
    result_list = []
    for a in listA:
        result_list.append(isPrime(a))
    return result_list

isPrime_list([1, 2, 3, 4, 5])
```

Output :

```
[False, True, True, False, True]
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ isPrime_list ที่รับ parameter เป็น listA
บรรทัดที่ 2 : สร้างตัวแปรชื่อ result_list และกำหนดให้เป็น empty list
บรรทัดที่ 3 : พิจารณาสมาชิกแต่ละตัวใน listA
บรรทัดที่ 4 : ตรวจสอบจำนวนเฉพาะโดยเรียกใช้ isPrime(a)
แล้วนำผลลัพท์ไป append ใน result_list
บรรทัดที่ 5 : return result_list ออกมา

เฉลยข้อที่ 35.

Input :

```
1  def draw_tree(n):
2      for i in range(n):
3          for j in range(n):
4              if (i==j) or (i+j == n-1):
5                  print('x', end='')
6              else:
7                  print('-', end='')
8          print()
9
10 draw_tree(5)
```

Output :

```
x---x
-x-x-
--x--
-x-x-
x---x
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ draw_tree ที่รับ parameter เป็น n
บรรทัดที่ 2-8 : เหมือนบท 12 ข้อ 106

เฉลยข้อที่ 36.

Input :

```
1 def draw_tree2(n):
2     if n%2 == 0:
3         for i in range(n-1):
4             for j in range(n):
5                 if (j==abs(int((n-1)/2)-i)) or (j==n
    ↪ -1-abs(int((n-1)/2)-i)):
6                     print('x', end='')
7                 else:
8                     print('-', end='')
9             print()
10    else:
11        for i in range(n):
12            for j in range(n):
13                if (j==abs((n-1)/2-i)) or (j==n-1-
    ↪ abs((n-1)/2-i)):
14                    print('x', end='')
15                else:
16                    print('-', end='')
17            print()
18
19 draw_tree2(5)
```

Output :

```
--x--
-x-x-
x---x
-x-x-
--x--
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ draw_tree2 ที่รับ parameter เป็น n

บรรทัดที่ 2-17 : เหมือนบท 12 ข้อ 107

13

เฉลยข้อที่ 37.

Input :

```
1  def draw_tree3():
2      for i in range(4):
3          for j in range(4+i):
4              if j < 3-i:
5                  print('-', end='')
6              else:
7                  print('x', end='')
8          print()
9
10 draw_tree3()
```

Output :

```
---x
--xxx
-xxxxx
xxxxxxx
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ draw_tree3 ที่รับ parameter เป็น n
บรรทัดที่ 2-17 : เหมือนบท 12 ข้อ 110

เฉลยข้อที่ 38.

Input :

```
def plus_and_cross(listA, listB):
    plus_list = []
    cross_list = []
    n = len(listA)
    for i in range(n):
        plus_list.append(listA[i] + listB[i])
        cross_list.append(listA[i]*listB[i])
    return plus_list, cross_list

plus_list, cross_list = plus_and_cross([1, 2, 3],
    ↪ [6, 7, 8])
print(plus_list)
print(cross_list)
```

Output :

```
[7, 9, 11]
[6, 14, 24]
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ plus_and_cross ที่รับ parameter เป็น listA และ listB
บรรทัดที่ 2-3 : สร้างตัวแปรชื่อ plus_list และ plus_list และกำหนดให้เป็น empty list สำหรับเก็บผลบวกและผลคูณตามลำดับ
บรรทัดที่ 4 : สร้างตัวแปรชื่อ n และกำหนดค่าเท่ากับความยาวของ listA
บรรทัดที่ 5 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 6 : หาผลบวกของสมาชิกตำแหน่งที่ i ของ listA และ listB แล้วนำไป append ใน plus_list
บรรทัดที่ 7 : หาผลคูณของสมาชิกตำแหน่งที่ i ของ listA และ listB แล้วนำไป append ใน cross_list
บรรทัดที่ 8 : return plus_list และ cross_list ออกมา

เฉลยข้อที่ 39.

Input :

```python
def key_and_value(dictA):
    key = dictA.keys()
    value = dictA.values()
    return key, value

key, value = key_and_value({'name' : 'John', 'age' :
    ↪ 32})
print(key)
print(value)
```


Output :

```
dict_keys(['name', 'age'])
dict_values(['John', 32])
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ key_and_value ที่รับ parameter เป็น dictA
บรรทัดที่ 2 : สร้างตัวแปรชื่อ key เพื่อเก็บ key ของ dictA
บรรทัดที่ 3 : สร้างตัวแปรชื่อ value เพื่อเก็บ value ของ dictA
บรรทัดที่ 4 : return key และ value ออกมา

เฉลยข้อที่ 40.

Input :

```
1  def find_cusum(listA):
2      cusum = 0
3      cusum_list = []
4      n = len(listA)
5      for i in range(n):
6          cusum = cusum + listA[i]
7          cusum_list.append(cusum)
8      return cusum_list
9
10 find_cusum([8, 0, -2, 4])
```

Output :

```
[8, 8, 6, 10]
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ find_cusum ที่รับ parameter เป็น listA
บรรทัดที่ 2 : สร้างตัวแปรชื่อ cusum และกำหนดค่าเท่ากับ 0
บรรทัดที่ 3 : สร้างตัวแปรชื่อ cusum_list และกำหนดให้เป็น empty list สำหรับเก็บค่าผลบวกสะสม
บรรทัดที่ 4 : สร้างตัวแปรชื่อ n และกำหนดค่าเท่ากับความยาวของ listA
บรรทัดที่ 5 : พิจารณาค่า i ตั้งแต่ 0 ถึง n-1
บรรทัดที่ 6 รวมผลบวกสะสมไว้ใน cusum
บรรทัดที่ 7 : นำผลบวกสะสมไป append ใน cusum_list
บรรทัดที่ 8 : return cusum_list ออกมา

เฉลยข้อที่ 41.

Input :

```
1 def isTriangle(a, b, c):
2     s_len = sorted([a, b, c])
3     if s_len[0] + s_len[1] > s_len[2]:
4         return True
5     else:
6         return False
7
8 isTriangle(5, 3, 4)
9 solution41(team)
```

Output :

```
True
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ isTriangle ที่รับ parameter เป็น a, b และ c
บรรทัดที่ 2 : สร้างตัวแปรชื่อ s_len เพื่อเรียงลำดับค่าของ a, b และ c จากน้อยไปมาก
บรรทัดที่ 3 : ทำการตรวจสอบว่าผลบวกของ 2 ด้านที่สั้นกว่า มีค่ามากกว่า ด้านที่ยาวที่สุดหรือไม่
บรรทัดที่ 4 : ถ้าเป็นจริง ก็จะครอบคลุมทุกเงื่อนไขของการเป็นด้านของสามเหลี่ยม return True
บรรทัดที่ 5-6 : ถ้าเป็นเท็จ return False

เฉลยข้อที่ 42.

Input :

```
def switch_case(word):
    sc_word = ''
    for c in word:
        if c == c.lower():
            sc_word = sc_word + c.upper()
        else:
            sc_word = sc_word + c.lower()
    return sc_word

switch_case('My Name Is John Doe')
```

Output :

```
'mY nAME iS jOHN dOE'
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ switch_case ที่รับ parameter เป็น word
บรรทัดที่ 2 : สร้างตัวแปรชื่อ sc_word และกำหนดให้เป็น empty string
บรรทัดที่ 3 : พิจารณาอักขระแต่ละตัวใน word
บรรทัดที่ 4-5 : ถ้าอักขระเป็น lower case จะนำ upper case ของอักขระไปใส่ใน sc_word
บรรทัดที่ 6-7 : ถ้าอักขระเป็น upper case จะนำ lower case ของอักขระไปใส่ใน sc_word
บรรทัดที่ 8 : return sc_word ออกมา

เฉลยข้อที่ 43.

Input :

```
1 def del_exceed_len(listA, n):
2     remain_list = []
3     for a in listA:
4         if len(a) <= n:
5             remain_list.append(a)
6     return remain_list
7
8 del_exceed_len(['Hello', 'Hel', 'Hell', 'He', 'H'],
    ↪ 3)
```

Output :

```
['Hel', 'He', 'H']
```


เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ del_exceed_len ที่รับ parameter เป็น listA และ n
บรรทัดที่ 2 : สร้างตัวแปรชื่อ remain_list และกำหนดให้เป็น empty list
บรรทัดที่ 3 : พิจารณาสมาชิกแต่ละตัวใน A
บรรทัดที่ 4-5 : ถ้าความยาวของสมาชิกน้อยกว่าหรือเท่ากับ n ก็จะนำไป append ใน remain_list
บรรทัดที่ 6 : return remain_list ออกมา

เฉลยข้อที่ 44.

Input :

```
1 def derivative(listA):
2     n = len(listA)
3     diff = [0]*n
4     for i in range(1, n):
5         diff[i-1] = listA[i]*i
6     return diff
7
8 derivative([2, 3, 0, 1])
```

Output :

```
[3, 0, 3, 0]
```

เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ derivative ที่รับ parameter เป็น listA
บรรทัดที่ 2 : สร้างตัวแปรชื่อ n และกำหนดค่าเท่ากับความยาวของ listA
บรรทัดที่ 3 : สร้างตัวแปรชื่อ diff และกำหนดให้เป็น list ที่มีค่าเริ่มต้นเป็น 0 และ ยาวเท่ากับ n
บรรทัดที่ 4 : พิจารณา index ใน A โดยเริ่มจาก index 1 จนถึงสุดท้าย เนื่องจาก index 0 จะทำการ diff แล้วมีค่าเป็น 0 เสมอ เพราะเป็นค่าคงที่
บรรทัดที่ 5 : เก็บสัมประสิทธิ์ของค่า derivatives ใส่ใน diff
บรรทัดที่ 6 : return diff ออกมา

เฉลยข้อที่ 45.

Input :

```
def reshape(listA, dim):
  old_n = len(listA)
  new_n = dim[0]*dim[1]
  matrixA = []
  if old_n == new_n :
    row = dim[0]
    col = dim[1]
    for i in range(row):
      matrixA.append(listA[i*col:(i+1)*col])
    return matrixA
  else :
    print('invalid dimension')

reshape([1,2,3,4,5,6,7,8,9,10,11,12], (4,3))
```

Output :

```
[[1, 2, 3], [4, 5, 6], [7, 8, 9], [10, 11, 12]]
```

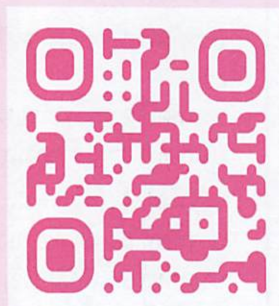
เฉลยละเอียด

บรรทัดที่ 1 : ประกาศฟังก์ชันชื่อ reshape ที่รับ parameter เป็น listA และ dim
บรรทัดที่ 2 : สร้างตัวแปรชื่อ old_n และกำหนดค่าเท่ากับความยาวของ listA
บรรทัดที่ 3 : สร้างตัวแปรชื่อ new_n เพื่อเก็บค่าผลคูณของสมาชิกใน dim
บรรทัดที่ 4 : สร้างตัวแปรชื่อ matrixA และกำหนดให้เป็น empty list
บรรทัดที่ 5 : ถ้า old_n เท่ากับ new_n ให้ทำบรรทัดที่ 6-10
บรรทัดที่ 6 : เก็บสมาชิกตำแหน่ง 0 ของ dim ไว้ใน row
บรรทัดที่ 7 : เก็บสมาชิกตำแหน่ง 1 ของ dim ไว้ใน col
บรรทัดที่ 8 : พิจารณาค่า i ตั้งแต่ 0 ถึง row-1
บรรทัดที่ 9 : นำ slice ของ listA ไป append ใน matrixA
บรรทัดที่ 10 : return matrixA ออกมา
บรรทัดที่ 11-12 : ถ้า old_n ไม่เท่ากับ new_n ให้พิมพ์ 'invalid dimension'

13

14 OOP


สแกนเพื่อเรียนวิดีโอเนื้อหา
และเฉลยแบบฝึกหัดบทที่ 14


สแกนเพื่อรับ Code บทที่ 14

14.1 Problem

จงเขียน code ของโจทย์ต่อไปนี้

1. เขียนโปรแกรมสร้าง class ชื่อ People โดยมี attribute และ method ดังนี้

attribute

- name เป็นชื่อของบุคคล
- age เป็นอายุของบุคคล

method

- introduce() เมื่อเรียกใช้จะพิมพ์ข้อความ My name is <name>. I'm <age> years old

ตัวอย่างที่ 1 :

Input :

```
John = People('John', 23)
John.introduce()
```

Output :

```
My name is John. I'm 23 years old.
```

2. เขียนโปรแกรมสร้าง class ชื่อ Human โดยมี attribute และ method ดังนี้

attribute

- name เป็นชื่อของบุคคล
- age เป็นอายุของบุคคล

method

- aging(years) รับ parameter 1 ตัว คือ years
  - แสดงอายุปัจจุบัน
  - เพิ่มอายุขึ้นเท่ากับ years
  - แสดงอายุหลังเพิ่มแล้ว

ตัวอย่างที่ 1 :

Input :

```
John = Human('John', 23)
John.aging(10)
```

Output :

```
Before : 23 years old
After : 33 years old
```

3. เขียนโปรแกรมสร้าง class ชื่อ ComplexNumber โดยมี attribute และ method ดังนี้

attribute

- r เป็นค่าของส่วนจริง
- i เป็นค่าของส่วนจินตภาพ

method

- get_value() ทำหน้าที่แสดงค่า cartesian form เช่น 2+3i, 5-4i เป็นต้น โดยเมื่อเรียกใช้ get_value() โปรแกรมจะพิมพ์ <r> + <i>i

ตัวอย่างที่ 1 :

Input :

```
Z = ComplexNumber(2, 4)
Z.get_value()
```

Output :

```
2+4i
```

4. เขียนโปรแกรมสร้าง class ชื่อ Circle โดยมี attribute และ method ดังนี้

attribute

- radius เป็นรัศมีของวงกลม

method

- get_area() ทำหน้าที่แสดงค่าพื้นที่ของวงกลม (area = pi*radius**2)
- get_perimeter() ทำหน้าที่แสดงค่าความยาวเส้นรอบวง (perimeter = 2*pi*radius)

ตัวอย่างที่ 1 :

Input :

```
circle_A = Circle(32)
circle_B = Circle(16)
print('Area Circle A :', circle_A.get_area())
print('Perimeter Circle A :', circle_A.
    ↪ get_perimeter())
print('Area Circle B :', circle_B.get_area())
print('Perimeter Circle B :', circle_B.
    ↪ get_perimeter())
```

Output :

```
Area Circle A : 3216.990877275948
Perimeter Circle A : 201.06192982974676
Area Circle B : 804.247719318987
Perimeter Circle B : 100.53096491487338
```

14

5. เขียนโปรแกรมสร้าง class ชื่อ Book โดยมี attribute และ method ดังนี้

attribute

- book_name เป็นชื่อของหนังสือ
- book_status เป็นสถานะการยืมของหนังสือ โดย book_status มีค่า default เป็น 'available'

method

- borrow_book()
  - ถ้า book_status เป็น 'available'
    - ให้พิมพ์ 'borrow successfully'
    - เปลี่ยน book_status เป็น 'unavailable'
  - แต่ถ้า เรียกใช้ตอนที่ book_status เป็น 'unavailable'
    - ให้พิมพ์ 'borrow unsuccessfully'
- return_book()
  - ถ้า book_status เป็น 'unavailable'
    - ให้พิมพ์ 'return successfully'
    - เปลี่ยน book_status เป็น 'available'
  - แต่ถ้า เรียกใช้ตอนที่ book_status เป็น 'available'
    - ให้พิมพ์ 'return unsuccessfully'

ตัวอย่างที่ 1 :

Input :

```
Book_A = Book('AI Superpowers')
print('Book status :', Book_A.book_status)
Book_A.borrow_book()
print('Book status :', Book_A.book_status)
Book_A.return_book()
print('Book status :', Book_A.book_status)
```

Output :

```
Book status : available
borrow successfully
Book status : unavailable
return successfully
Book status : available
```

14

6. เขียนโปรแกรมสร้าง class ชื่อ Car โดยมี attribute และ method ดังนี้
attribute

- brand เป็นยี่ห้อของรถ
- model เป็นรุ่นของรถ
- year เป็นปีของรถ
- color เป็นสีของรถ

method

- new_color(color) รับพารามิเตอร์ 1 ตัว ชื่อว่า color ทำหน้าที่แสดงสีใหม่ของรถที่ต้องการเปลี่ยน

ตัวอย่างที่ 1 :

Input :

```
car_A = Car('Honda', 'Civic', '2019', 'Black')
car_A.new_color('Red')
print('color =', car_A.color)
```

Output :

```
color = Red
```

7. เขียนโปรแกรมสร้าง class ชื่อ Dog โดยมี attribute และ method ดังนี้
attribute

- breed เป็นพันธุ์สุนัข
- color เป็นสีของสุนัข
- height เป็นส่วนสูง (หน่วยเซนติเมตร)
- weight เป็นน้ำหนัก (หน่วยกิโลกรัม)

method

- growth() ทำหน้าที่แสดงความสูงและน้ำหนักที่เพิ่มขึ้นของสุนัขอย่างละ 10%

ตัวอย่างที่ 1 :

Input :

```
dog_A = Dog('Jack Russell Terrier', 'White',
    ↪ 30, 7)
dog_A.growth()
print('height =', dog_A.height)
print('weight =', dog_A.weight)
```

Output :

```
height = 33.0
weight = 7.700000000000001
```

8. เขียนโปรแกรมสร้าง class ชื่อ Stats โดยมี attribute และ method ดังนี้

attribute

- data เป็น list ที่เก็บจำนวนใด ๆ

method

- mean() ทำหน้าที่คำนวณค่าเฉลี่ยของจำนวนใน data
- min() ทำหน้าที่คำนวณค่าต่ำสุดของจำนวนใน data
- max() ทำหน้าที่คำนวณค่าสูงสุดของจำนวนใน data
- summary() ทำหน้าที่แสดงค่า mean, min และ max โดยเมื่อเรียกใช้โปรแกรมจะพิมพ์ Mean : <mean>, Min : <min> และ Max : <max> ออกมา

ตัวอย่างที่ 1 :

Input :

```
A = [1,3,5,7,9,11]
stat_A = Stats(A)
stat_A.summary()
```

Output :

```
Mean : 6.0
Min : 1
Max : 11
```

14

9. เขียนโปรแกรมสร้าง class ชื่อ USDTHB โดยมี attribute และ method ดังนี้
attribute

- amount เป็นจำนวนเงิน
- currency เป็นสกุลเงิน

method

- to_THB() ทำหน้าที่แปลงสกุลเงินไปเป็น THB
- to_USD() ทำหน้าที่แปลงสกุลเงินไปเป็น USD

(กำหนดให้ใช้อัตราแลกเปลี่ยน 1 USD = 30 THB)

ตัวอย่างที่ 1 :

Input :

```
money = USDTHB(30000, 'USD')
print(money.amount, money.currency)
money.to_THB()
print(money.amount, money.currency)
money.to_USD()
print(money.amount, money.currency)
```

Output :

```
30000 USD
900000 THB
30000.0 USD
```

14

10. เขียนโปรแกรมสร้าง class ชื่อ Cashier โดยมี attribute และ method ดังนี้
attribute

- products เป็น list ที่เก็บชื่อสินค้า

method

- recommend() ทำหน้าที่แนะนำสินค้า โดยเมื่อเรียกใช้ โปรแกรมจะพิมพ์ We have <product>. ออกมา

ตัวอย่างที่ 1 :

Input :

```
cashier = Cashier(['apple', 'banana', 'orange'
    ↪ ])
cashier.recommend()
```

Output :

```
We have apple, banana, orange.
```

11. เขียนโปรแกรมสร้าง class ชื่อ Warrior โดยมี attribute และ method ดังนี้

attribute

- power เป็นพลังโจมตี
- defense เป็นพลังป้องกัน
- HP เป็นพลังชีวิต

method

- attack(warrior_object) ทำหน้าที่แสดงค่าพลังชีวิตหลังจากถูกโจมตี โดยที่
  - ถ้า power ≤ defense แล้ว HP จะไม่ลด
  - ถ้า power > defense แล้ว HP จะมีค่าเท่ากับ HP-(power-defense)

  โดยเมื่อเรียกใช้ โปรแกรมจะแสดงค่า HP ออกมา และถ้า HP = 0 ให้พิมพ์ Enemy died ออกมา

14

ตัวอย่างที่ 1 :

Input :

```
Warrior_A = Warrior(100,50,80)
Warrior_B = Warrior(60, 80, 120)
print('=== Before Attack ===')
print('Warrior_A HP =', Warrior_A.HP)
print('Warrior_B HP =', Warrior_B.HP)
Warrior_A.attack(Warrior_B)
Warrior_B.attack(Warrior_A)
print('=== After Attack ===')
print('Warrior_A HP =', Warrior_A.HP)
print('Warrior_B HP =', Warrior_B.HP)
```

Output :

```
=== Before Attack ===
Warrior_A HP = 80
Warrior_B HP = 120
=== After Attack ===
Warrior_A HP = 70
Warrior_B HP = 100
```

12. เขียนโปรแกรมสร้าง class Point โดยมี attribute และ method ดังนี้

attribute

- x เป็นค่าตามแกน x ในระนาบ 2 มิติ
- y เป็นค่าตามแกน y ในระนาบ 2 มิติ

method

- move_up(d) รับ parameter 1 ตัว คือ d
  - เลื่อนตำแหน่งขึ้นตามแนวแกน y ไป d หน่วย
  - พิมพ์ Current position : (<x>, <y>) ออกมา
- move_down(d) รับ parameter 1 ตัว คือ d
  - เลื่อนตำแหน่งลงตามแนวแกน y ไป d หน่วย
  - พิมพ์ Current position : (<x>, <y>) ออกมา
- move_left(d) รับ parameter 1 ตัว คือ d
  - เลื่อนตำแหน่งไปทางซ้ายตามแนวแกน x ไป d หน่วย
  - พิมพ์ Current position : (<x>, <y>) ออกมา

- move_right(d) รับ parameter 1 ตัว คือ d
  - เลื่อนตำแหน่งไปทางขวาตามแนวแกน x ไป d หน่วย
  - พิมพ์ Current position : (<x>, <y>) ออกมา

ตัวอย่างที่ 1 :

Input :

```
A = Point(0,0)
A.move_up(20)
A.move_left(23)
A.move_down(65)
A.move_right(50)
```

Output :

```
Current position : (0,0)
Current position : (0,20)
Current position : (-23,20)
Current position : (-23,-45)
Current position : (27,-45)
```

13. เขียนโปรแกรมสร้าง class ชื่อ Businessman โดยมี attribute และ method ดังนี้

attribute

- HP เป็นพลังชีวิต
- money เป็นพลังการเงิน
- happiness เป็นพลังความสุข

method

- work() ทำหน้าที่แสดงค่าพลังชีวิต การเงิน และความสุข หลังจากผ่านการทำงาน โดยจะเพิ่ม money 20 หน่วย และ happiness 10 หน่วย แต่จะลด HP 10 หน่วย
- sleep() ทำหน้าที่แสดงค่าพลังชีวิต การเงิน และความสุข หลังจากผ่านการนอนหลับ โดยจะเพิ่ม HP 20 หน่วย และ happiness 10 หน่วย แต่จะลด money 10 หน่วย
- play() ทำหน้าที่แสดงค่าพลังชีวิต การเงิน และความสุข หลังจากผ่านการเล่นเกม โดยจะลด HP และ money 10 หน่วย แต่จะเพิ่ม happiness 20 หน่วย

ตัวอย่างที่ 1 :

Input :

```
John = Businessman(100,100,100)
John.work()
John.report()
John.play()
John.report()
John.sleep()
John.report()
```

Output :

```
HP = 100, Money = 100, Happiness = 100
HP = 90, Money = 120, Happiness = 110
HP = 80, Money = 110, Happiness = 130
HP = 100, Money = 100, Happiness = 140
```

14. เขียนโปรแกรมสร้าง class ชื่อ Driver โดยมี attribute และ method ดังนี้

attribute

- HP เป็นพลังชีวิต
- generated_money เป็นเงินที่หาได้

method

- drive() ทำหน้าที่แสดงค่าพลังชีวิต และเงินที่หามาได้ หลังจากขับแท็กซี่ โดยจะลด HP 10 หน่วย แต่จะเพิ่ม generated_money 10 หน่วย
- care() ทำหน้าที่แสดงค่าพลังชีวิต และเงินที่หามาได้ หลังจากพักผ่อน โดยจะเพิ่ม HP 10 หน่วย แต่จะลด generated_money 10 หน่วย

ตัวอย่างที่ 1 :

Input :

```
driver_A = Driver(100,100)
driver_A.drive()
driver_A.report()
driver_A.care()
driver_A.report()
```

Output :

```
HP = 90, Generated Money = 110
HP = 100, Generated Money = 100
```

15. เขียนโปรแกรมสร้าง class Tree โดยมี attribute และ method ดังนี้
attribute

- height เป็นความสูงต้นไม้
- width เป็นความกว้างต้นไม้
- generated_money เป็นเงินที่หาได้

method

- feed_A() ทำหน้าที่แสดงค่าเงินที่หาได้ และความกว้างของต้นไม้หลังจากให้ปุ๋ยชนิด A โดยจะลด generated_money 10 หน่วย แต่จะเพิ่ม width 12 หน่วย
- feed_B() ทำหน้าที่แสดงค่าเงินที่หาได้ และความสูงของต้นไม้หลังจากให้ปุ๋ยชนิด B โดยจะลด generated_money 8 หน่วย แต่จะเพิ่ม height 10 หน่วย
- sell() ทำหน้าที่แสดงค่าจำนวนเงินที่ขายต้นไม้ได้ โดย generated_money จะเพิ่มเท่ากับ width*height*0.8

ตัวอย่างที่ 1 :

Input :

```
tree_A = Tree(10,10, 1000)
tree_A.feed_A()
tree_A.feed_B()
tree_A.sell()
```

Output :

```
Width = 22, Height = 20
Generated money = 1223.683319527346
```

14

14.2 Solution

เฉลยข้อที่ 1.

Input :

```
class People :
  def __init__(self, name, age):
    self.name = name
    self.age = age

  def introduce(self):
    print('My name is', self.name, end='.')
    print("I'm", self.age, 'years old.')

John = People('John', 23)
John.introduce()
```

Output :

```
My name is John. I'm 23 years old.
```

เฉลยข้อที่ 2.

Input :

```
class Human :
  def __init__(self, name, age):
    self.name = name
    self.age = age

  def aging(self, years) :
    print('Before :', self.age, 'years old')
    self.age = self.age + years
    print('After :', self.age, 'years old')

John = Human('John', 23)
John.aging(10)
```

Output :

```
Before : 23 years old
After : 33 years old
```

14

เฉลยข้อที่ 3.

Input :

```
class ComplexNumber :
  def __init__(self, r, i):
    self.r = r
    self.i = i

  def get_value(self):
    print('%d + %di' %(self.r, self.i))

Z = ComplexNumber(2, 4)
Z.get_value()
```

Output :

```
2+4i
```

เฉลยข้อที่ 4.

Input :

```
class Circle :
  import math
  def __init__(self, radius):
    self.radius = radius

  def get_area(self):
    return math.pi*self.radius**2

  def get_perimeter(self):
    return 2*math.pi*self.radius

circle_A = Circle(32)
circle_B = Circle(16)
print('Area Circle A :', circle_A.get_area())
print('Perimeter Circle A :', circle_A.get_perimeter
    ↪ ())
print('Area Circle B :', circle_B.get_area())
print('Perimeter Circle B :', circle_B.get_perimeter
    ↪ ())
```

14

Output :

```
Area Circle A : 3216.990877275948
Perimeter Circle A : 201.06192982974676
Area Circle B : 804.247719318987
Perimeter Circle B : 100.53096491487338
```

เฉลยข้อที่ 5.

Input :

```
class Book :
  def __init__(self, book_name, book_status='
    ↪ available'):
    self.book_name = book_name
    self.book_status = book_status

  def borrow_book(self):
    if self.book_status == 'available':
      print('borrow successfully')
      self.book_status = 'unavailable'
    else :
      print('borrow unsuccessfully')

  def return_book(self):
    if self.book_status == 'unavailable':
      print('return successfully')
      self.book_status = 'available'
    else :
      print('return unsuccessfully')

Book_A = Book('AI Superpowers')
print('Book status :', Book_A.book_status)
Book_A.borrow_book()
print('Book status :', Book_A.book_status)
Book_A.return_book()
print('Book status :', Book_A.book_status)
```

Output :

```
Book status : available
borrow successfully
Book status : unavailable
return successfully
Book status : available
```

เฉลยข้อที่ 6.

Input :

```
class Car :
  def __init__(self, brand, model, year, color):
    self.brand = brand
    self.model = model
    self.year = year
    self.color = color

  def new_color(self, color) :
    self.color = color

car_A = Car('Honda', 'Civic', '2019', 'Black')
car_A.new_color('Red')
print('color =', car_A.color)
```

Output :

```
color = Red
```

เฉลยข้อที่ 7.

Input :

```
class Dog :
  def __init__(self, breed, color, height, weight):
    self.breed = breed
    self.color = color
    self.height = height
    self.weight = weight

  def growth(self):
    self.height = self.height*1.1
    self.weight = self.weight*1.1

dog_A = Dog('Jack Russell Terrier', 'White', 30, 7)
dog_A.growth()
print('height =', dog_A.height)
print('weight =', dog_A.weight)
```

Output :

```
height = 33.0
weight = 7.700000000000001
```

เฉลยข้อที่ 8.

Input :

```
class Stats :
  def __init__ (self, data):
    self.data = data

  def mean(self):
    mean = sum(self.data) / len(self.data)
    return mean

  def min(self):
    _min = min(self.data)
    return _min

  def max(self):
    _max = max(self.data)
    return _max

  def summary(self):
    print('Mean :', self.mean())
    print('Min :', self.min())
    print('Max :', self.max())

A = [1,3,5,7,9,11]
stat_A = Stats(A)
stat_A.summary()
```

Output :

```
Mean : 6.0
Min : 1
Max : 11
```

เฉลยข้อที่ 9.

Input :

```
class USDTHB :
  def __init__(self, amount, currency):
    self.amount = amount
    self.currency = currency

  def to_USD(self):
    if self.currency == 'THB' :
      self.currency = 'USD'
      self.amount = self.amount / 30
      return self.amount / 30
    else :
      return self.amount

  def to_THB(self):
    if self.currency == 'USD' :
      self.currency = 'THB'
      self.amount = self.amount * 30
      return self.amount * 30
    else :
      return self.amount

money = USDTHB(30000, 'USD')
print(money.amount, money.currency)
money.to_THB()
print(money.amount, money.currency)
money.to_USD()
print(money.amount, money.currency)
```


Output :

```
30000 USD
900000 THB
30000.0 USD
```

เฉลยข้อที่ 10.

Input :

```
class Cashier :
  def __init__(self, products) :
    self.products = products

  def recommend(self) :
    str_product = ', '.join(self.products)
    print('We have '+str(str_product)+'.')

cashier = Cashier(['apple', 'banana', 'orange'])
cashier.recommend()
```

Output :

```
We have apple, banana, orange.
```

เฉลยข้อที่ 11.

Input :

```
class Warrior :
  def __init__(self, power, defense, HP) :
    self.power = power
    self.defense = defense
    self.HP = HP

  def attack(self, enemy) :
    if enemy.HP > 0 and self.HP > 0:
      enemy.HP = max(0, enemy.HP - max(0, self.power
    ↪  - enemy.defense))
    if enemy.HP == 0 :
      print('Enemy died')

Warrior_A = Warrior(100,50,80)
Warrior_B = Warrior(60, 80, 120)
print('=== Before Attack ===')
print('Warrior_A HP =', Warrior_A.HP)
print('Warrior_B HP =', Warrior_B.HP)
Warrior_A.attack(Warrior_B)
Warrior_B.attack(Warrior_A)
print('=== After Attack ===')
print('Warrior_A HP =', Warrior_A.HP)
print('Warrior_B HP =', Warrior_B.HP)
```

Output :

```
=== Before Attack ===
Warrior_A HP = 80
Warrior_B HP = 120
=== After Attack ===
Warrior_A HP = 70
Warrior_B HP = 100
```

เฉลยข้อที่ 12.

Input :

```
class Point :
  def __init__(self, x, y):
    self.x = x
    self.y = y
    print(f'Current position : ({self.x},{self.y})')

  def move_up(self, d):
    self.y = self.y + d
    print(f'Current position : ({self.x},{self.y})')

  def move_down(self, d):
    self.y = self.y - d
    print(f'Current position : ({self.x},{self.y})')

  def move_left(self, d):
    self.x = self.x - d
    print(f'Current position : ({self.x},{self.y})')

  def move_right(self, d):
    self.x = self.x + d
    print(f'Current position : ({self.x},{self.y})')

A = Point(0,0)
A.move_up(20)
A.move_left(23)
A.move_down(65)
A.move_right(50)
```

14

Output :

```
Current position : (0,0)
Current position : (0,20)
Current position : (-23,20)
Current position : (-23,-45)
Current position : (27,-45)
```

เฉลยข้อที่ 13.

Input :

```
class Businessman :
  def __init__(self, HP, money, happiness):
    self.HP = HP
    self.money = money
    self.happiness = happiness
    self.report()

  def work(self):
    self.money = self.money + 20
    self.happiness = self.happiness + 10
    self.HP = self.HP - 10

  def sleep(self):
    self.money = self.money - 10
    self.HP = self.HP + 20
    self.happiness = self.happiness + 10

  def play(self):
    self.happiness = self.happiness + 20
    self.HP = self.HP - 10
    self.money = self.money - 10

  def report(self):
    print('HP = %d, Money = %d, Happiness = %d' %(
    ↪ self.HP, self.money, self.happiness))

John = Businessman(100,100,100)
John.work()
John.report()
John.play()
John.report()
John.sleep()
John.report()
```

14

Output :

```
HP = 100, Money = 100, Happiness = 100
HP = 90, Money = 120, Happiness = 110
HP = 80, Money = 110, Happiness = 130
HP = 100, Money = 100, Happiness = 140
```

เฉลยข้อที่ 14.

Input :

```
class Driver :
  def __init__(self, HP, generated_money):
    self.HP = HP
    self.generated_money = generated_money

  def drive(self):
    self.HP = self.HP - 10
    self.generated_money = self.generated_money + 10

  def care(self):
    self.HP = self.HP + 10
    self.generated_money = self.generated_money - 10

  def report(self):
    print('HP = ', self.HP, ', Generated Money = ',
    ↪ self.generated_money)

driver_A = Driver(100,100)
driver_A.drive()
driver_A.report()
driver_A.care()
driver_A.report()
```

Output :

```
HP = 90, Generated Money = 110
HP = 100, Generated Money = 100
```

เฉลยข้อที่ 15.

Input :

```
class Tree :
  def __init__(self, height, width, generated_money)
    ↪ :
    self.height = height
    self.width = width
    self.generated_money = generated_money

  def feed_A(self) :
    self.generated_money = self.generated_money - 10
    self.width = self.width + 12

  def feed_B(self) :
    self.generated_money = self.generated_money - 8
    self.height = self.height + 10

  def sell(self) :
    self.generated_money = self.generated_money +
    ↪ self.width * self.height ** 0.8
    print('Width =', self.width, 'Height =', self.
    ↪ height)
    print('Generated money =', self.generated_money)

tree_A = Tree(10,10, 1000)
tree_A.feed_A()
tree_A.feed_B()
tree_A.sell()
```


Output :

```
Width = 22, Height = 20
Generated money = 1223.683319527346
```

“

หนังสือเล่มนี้ถูกออกแบบมาอย่างพิถีพิถัน
เพื่อให้ง่ายต่อการทำความเข้าใจภาษา Python มากที่สุด
เท่าที่จะเป็นไปได้ ผ่านองค์ประกอบเสริมในเล่ม
ที่จะช่วยให้เกิดการเรียนรู้อย่างครบวงจร

ทีมงาน Tautology

”

องค์ประกอบในหนังสือเล่มนี้ได้แก่

- วิดีโอสอนการเขียน Python พร้อมอธิบายแบบฝึกหัด ความยาวกว่า 30 ชั่วโมงบน Youtube Chanel
- สไลด์ประกอบการสอน เนื้อหาอัดแน่นกว่า 250 หน้า
- Source Code เฉลยของโจทย์ทุกข้อใน Jupyter Notebook
- Facebook group สำหรับสอบถามข้อมูล และแลกเปลี่ยนความคิดเห็น
- ลิงก์รวบรวมตำแหน่งงานในสายงานภาษา Python อัพเดทแบบ Real Time

(ทั้งหมดนี้ไม่มีการคิดค่าใช้จ่ายเพิ่มเติม)

ISBN 978-616-93753-0-2



Bht 495-.